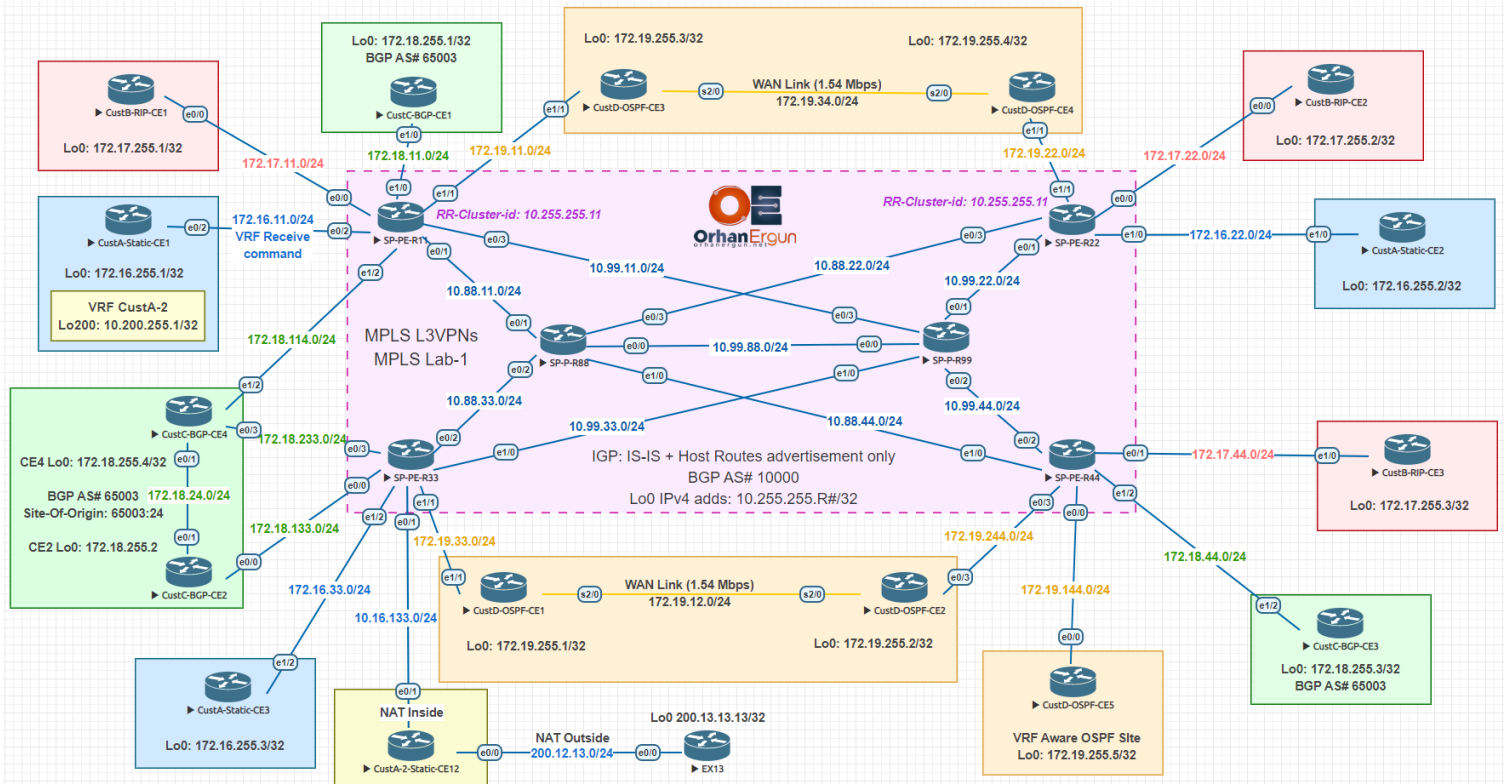


MPLS Lab 01

MPLS Layer 3 Virtual Private Networks

Topology:



BGP AF/SAF: VPNv4 Unicast

PE-CE Routing Protocols:

- Static Routing (Including VRF Receive command)
- RIP
- OSPF (Also includes Sham-link between PEs and VRF Aware OSPF process on CE5)
- BGP (Same Local AS for all Customer sites)

Task 01:

Assign IPv4 addresses to all interfaces in the SP network (Every inter-node connection prefix length should be /24 and IP addressing scheme should be 10.Y.X.R/24 where Y stands for highest R#, X stands for lowest R# and R stands for R#, for example the IP address of R88's interface between R88 and R11 should have IP address of 10.88.11.88/24) and also all SP routers should have a Lo0 interface with this scheme : 10.255.255.R/32 where R stands for router number, for example R99's Lo0 interface should have IPv4 address of 10.255.255.99/32.

- Create an IS-IS Process
- IS-IS NET scheme: 49.0000.0000.0000.00RN.00 (RN stands for router number)
- Configure the routers to only form Level-2 adjacency
- Advertise Lo0 interfaces in the IS-IS
- Configure adjacency logging for IS-IS
- Enable IS-IS 10000 on all SP router interfaces

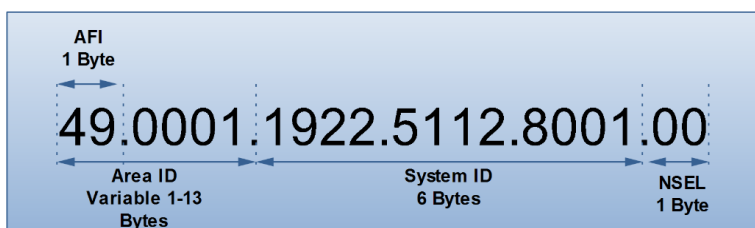
Solution:

You don't need to follow the exact procedure of tasks, just need to complete the task in any order you like! For example I first create IS-IS process:

SP-P-R88:

```
router isis
net 49.0000.0000.0000.0088.00
is-type level-2-only
log-adjacency-changes all
passive-interface Loopback0
!
```

Take a look at the NET command, you remember Router-ID in OSPF? It is the equivalent to OSPF Router-ID in IS-IS, not exactly but just think that way! There is 49.0000 in the beginning, it is the Area-ID, in this Lab we are going to use only a Single area IS-IS, then we configure 49.0000 for all routers NET.



Next we have System ID, every router in the domain should have a unique System ID, for SP-P-R88 router it is set to 0000.0000.0088.

Let NSEL to be 00 on all routers.

Next we should configure all routers to form only L2 Adjacency, if we don't configure this command all routers form L1 as well as L2 adjacency, and the task only wants us to configure only single area L2 adjacency. we can configure this adjacency type per interface, but in this case this is easier to configure L2-Adjacency only under IS-IS process.

By default IS-IS does not generate any syslog messages for Adjacency changes on Cisco routers (unlike OSPF), in order to enable logging for IS-IS we must put **log-adjacency-changes all** command under IS-IS process to get detailed logs.

The task asks us to also advertise Lo0 interfaces in IS-IS domain. IS-IS's behavior differs from OSPF, by making an interface passive in IS-IS, that interface is also being advertised in the IS-IS domain.

Let's configure other routers as well:

SP-P-R99:

```
router isis
 net 49.0000.0000.0000.0099.00
 is-type level-2-only
 log-adjacency-changes all
 passive-interface Loopback0
```

!

SP-PE-R11:

```
router isis
 net 49.0000.0000.0000.0011.00
 is-type level-2-only
 log-adjacency-changes all
 passive-interface Loopback0
```

!

SP-PE-R22:

```
router isis
 net 49.0000.0000.0000.0022.00
 is-type level-2-only
 log-adjacency-changes all
 passive-interface Loopback0
```

```
SP-PE-R33:
router isis
 net 49.0000.0000.0000.0033.00
 is-type level-2-only
 log-adjacency-changes all
 passive-interface Loopback0
!
SP-PE-R44:
router isis
 net 49.0000.0000.0000.0044.00
 is-type level-2-only
 log-adjacency-changes all
 passive-interface Loopback0
```

Now we need to configure IPv4 addresses on SP router's interfaces and also make the IS-IS network type point-to-point then enable IS-IS on those interfaces.

```
SP-P-R88:
interface Loopback0
 ip address 10.255.255.88 255.255.255.255
!
interface Ethernet0/0
 bandwidth 10000000
 ip address 10.99.88.88 255.255.255.0
 ip router isis
!
interface Ethernet0/1
 bandwidth 10000000
 ip address 10.88.11.88 255.255.255.0
 ip router isis
!
interface Ethernet0/2
 bandwidth 10000000
 ip address 10.88.33.88 255.255.255.0
 ip router isis
!
```

```
interface Ethernet0/3
  bandwidth 10000000
  ip address 10.88.22.88 255.255.255.0
  ip router isis
!
interface Ethernet1/0
  bandwidth 10000000
  ip address 10.88.44.88 255.255.255.0
  ip router isis
!
```

we issued the `bandwidth 10000000` (10 Gb/s) command to just pretend that these links are 10 Gb/s links in SP core, because this IOS does not support any 10 Mb/s+ links. this command does not affect router's interfaces speed, only it is used for QoS topics, Traffic Engineering, IGP's metric calculations and also affects EIGRP Bandwidth-limitation, so be careful using this explicit bandwidth command!

By entering `ip router isis` command, that interface would become part of IS-IS process and actively looks at neighboring routers to form adjacency and also that the network would be advertised in the IS-IS routing domain.

IS-IS is an L2 protocol, it has its own L2 Protocol and also uses DSAP/SSAP LLC fields, unlike OSPF and EIGRP it does not use L3 protocol. We have captured some traffics using WireShark, Take a look at below capture:

```
> Frame 2: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits) on interface 0
  > IEEE 802.3 Ethernet
    > Destination: ISIS-all-level-2-IS's (01:80:c2:00:00:15)
    > Source: aa:bb:cc:00:20:00 (aa:bb:cc:00:20:00)
    Length: 1500
  > Logical-Link Control
    > DSAP: ISO Network Layer (0xfe)
    > SSAP: ISO Network Layer (0xfe)
    > Control field: U, func=UI (0x03)
  > ISO 10589 ISIS INTRA Domain Routing Information Exchange Protocol
    Intradomain Routing Protocol Discriminator: ISIS (0x83)
    Length Indicator: 27
    Version/Protocol ID Extension: 1
    ID Length: 0
    000. .... = Reserved: 0x0
    ...1 0000 = PDU Type: L2 HELLO (16)
    Version: 1
    Reserved: 0
    Maximum Area Addresses: 0
  > ISIS HELLO
```

Let's configure other routers as well:

SP-P-R99:

```
interface Loopback0
 ip address 10.255.255.99 255.255.255.255
!
interface Ethernet0/0
 bandwidth 10000000
 ip address 10.99.88.99 255.255.255.0
 ip router isis
!
interface Ethernet0/1
 bandwidth 10000000
 ip address 10.99.22.99 255.255.255.0
 ip router isis
!
interface Ethernet0/2
 bandwidth 10000000
 ip address 10.99.44.99 255.255.255.0
 ip router isis
!
interface Ethernet0/3
 bandwidth 10000000
 ip address 10.99.11.99 255.255.255.0
 ip router isis
!
interface Ethernet1/0
 bandwidth 10000000
 ip address 10.99.33.99 255.255.255.0
 ip router isis
!
```

SP-PE-R11:

```
interface Loopback0
 ip address 10.255.255.11 255.255.255.255
!
```

```
interface Ethernet0/1
  bandwidth 10000000
  ip address 10.88.11.11 255.255.255.0
  ip router isis
!
```

```
interface Ethernet0/3
  bandwidth 10000000
  ip address 10.99.11.11 255.255.255.0
  ip router isis
!
```

SP-PE-R22:

```
interface Loopback0
  ip address 10.255.255.22 255.255.255.255
!
```

```
interface Ethernet0/1
  bandwidth 10000000
  ip address 10.99.22.22 255.255.255.0
  ip router isis
!
```

```
interface Ethernet0/3
  bandwidth 10000000
  ip address 10.88.22.22 255.255.255.0
  ip router isis
!
```

SP-PE-R33:

```
interface Loopback0
  ip address 10.255.255.33 255.255.255.255
!
```

```
interface Ethernet0/2
  bandwidth 10000000
  ip address 10.88.33.33 255.255.255.0
  ip router isis
!
```

```
interface Ethernet1/0
  bandwidth 10000000
  ip address 10.99.33.33 255.255.255.0
  ip router isis
!
```

SP-PE-R44:

```
interface Loopback0
  ip address 10.255.255.44 255.255.255.255
!
interface Ethernet0/2
  bandwidth 10000000
  ip address 10.99.44.44 255.255.255.0
  ip router isis
!
interface Ethernet1/0
  bandwidth 10000000
  ip address 10.88.44.44 255.255.255.0
  ip router isis
!
```




Task 02:

- Define MPLS label range for every router, so we can track packets LSPs easily in this lab, you don't have to always do that, we just do it for some purposes you will see that!
- Configure static router-id for LDP on every router and force it to use Lo0 interface as LDP Router-ID.
- Enable MPLS LDP session protection.
- Enable MPLS and LDP on all IS-IS enabled interfaces by using IS-IS Auto-Config mechanism.
- Enable MPLS LDP IGP Synchronization on all IS-IS enabled interfaces by using a single command.
- Configure IS-IS in order to only advertise host routes (/32 Networks of Lo0 interfaces)



Solution:

```
SP-P-R88:
mpls label range 80000 89999
mpls ldp session protection
mpls ldp router-id Loopback0
router isis
mpls ldp sync
mpls ldp autoconfig
advertise passive-only
!
```

MPLS LDP session protection feature provides faster LDP convergence when a link recovers following an outage. It helps routers to maintain LDP distributed Labels for a while till neighbor comes up again, so after the neighbor becomes alive no need to label generation and distribution between those neighbors for the same prefixes.

The task asked that we should configure Lo0 interface of every router as being used as LDP Router-ID. Two neighbors form LDP session using TCP port number 646 and also UDP port 646 to discover and track the state of neighbor using a hello mechanism, as you already know we need a source IP address and also a destination IP address to form a TCP session, Cisco router uses Router-ID as the source for LDP packets/datagrams, if we don't configure any explicit Router-ID for LDP, the IOS is going to use the highest non-shutdown Lo0 interface IP address as the source of its packets. It's a best practice to make this happen in a predictable way, maybe router chooses a Loopback address as a source for LDP sessions that we have not

advertised in IS-IS session. So we configure an explicit Router-id for all routers using **mpls ldp router-id lo0** command.

The task asked that we should configure Lo0 interface of every router as being used as LDP Router-ID. Two neighbors form LDP session using TCP port number 646 and also UDP port 646 to discover and track the state of neighbor using a hello mechanism, as you already know we need a source IP address and also a destination IP address to form a TCP session, Cisco router uses Router-ID as the source for LDP packets/datagrams, if we don't configure any explicit Router-ID for LDP, the IOS is going to use the highest non-shutdown Lo0 interface IP address as the source of its packets. It's a best practice to make this happen in a predictable way, maybe router chooses a Loopback address as a source for LDP sessions that we have not advertised in IS-IS session. So we configure an explicit Router-id for all routers using **mpls ldp router-id lo0** command.

By entering **mpls ldp sync** command under router isis process, MPLS LDP IGP Synchronization would be enabled on every IS-IS enabled interfaces (Physical interfaces), it prevents black holing issue.

When IGP is established between two routers, the router begins forwarding packets based on IGP learned routes (RIB/FIB) even before LDP Label exchange completes between the peers. Then the packets will be exchanged unlabeled then those packets will be black-holed because of label absence. The second problem is, if an LDP session closes, the device continues to forward traffic using the link associated with the LDP peer rather than an alternate pathway with a fully synchronized LDP session.

IGP LDP Sync feature prevents this packet loss. You will see in Segment-Routing, we don't face those kind of problems because we don't have to have a separate protocol like LDP, in SR, Index (let's say label) is also carried by IGP itself.

The task also asked us to enable MPLS on every physical interface. In order to do that we have two choices, the first one is to enable MPLS forwarding on each interface by using **mpls ip** interface-level command, the second and convenient one is to issue **mpls ldp autoconfig** command under IS-IS process. By using the second option MPLS forwarding is going to be enabled on each interface that we have enabled IS-IS on.

Task also asks us to advertise only /32 host routes. In MPLS we care only about reachability between two hosts not reachability of physical interfaces of those hosts! As far as one host is able to reach the other host's Lo0 address using it's Lo0 address we are happy! Because we will use BGP with different Address-Families or sometimes LDP in order to make MPLS beneficial. No one would want to form an iBGP session based on Physical interfaces, when we have Loopbacks that are always up because those are not physical interfaces, as long as we can reach Loopback interfaces somehow the iBGP sessions will be up. You will realize why when we do next tasks and all other MPLS Labs of the workbook.

The command `advertise passive-only` under router isis process suppresses the advertisement of other interface prefixes other than passive ones. You remember the previous task was asking us to make Loopback0 interfaces passive. You can achieve the same result in OSPF by using **prefix-suppression** command.

Let's configure other routers as well:

SP-P-R99:

```
mpls label range 90000 99999
mpls ldp session protection
mpls ldp router-id Loopback0
router isis
  mpls ldp sync
  mpls ldp autoconfig
  advertise passive-only
!
```

SP-PE-R11:

```
mpls label range 10000 19999
mpls ldp session protection
mpls ldp router-id Loopback0
router isis
  mpls ldp sync
  mpls ldp autoconfig
  advertise passive-only
!
```

SP-PE-R22:

```
mpls label range 20000 29999
mpls ldp session protection
mpls ldp router-id Loopback0
router isis
  mpls ldp sync
  mpls ldp autoconfig
  advertise passive-only
!
```

SP-PE-R33:

```

mpls label range 30000 39999
mpls ldp session protection
mpls ldp router-id Loopback0
router isis
  mpls ldp sync
  mpls ldp autoconfig
  advertise passive-only
!
```

SP-PE-R44:

```

mpls label range 40000 49999
mpls ldp session protection
mpls ldp router-id Loopback0
router isis
  mpls ldp sync
  mpls ldp autoconfig
  advertise passive-only
!
```



Verification:

```
SP-P-R88#show isis neighbors
```

System Id	Type	Interface	IP Address	State	Holdtime	Circuit Id
SP-PE-R11	L2	Et0/1	10.88.11.11	UP	7	SP-PE-R11.01
SP-PE-R22	L2	Et0/3	10.88.22.22	UP	9	SP-PE-R22.02
SP-PE-R33	L2	Et0/2	10.88.33.33	UP	8	SP-PE-R33.01
SP-PE-R44	L2	Et1/0	10.88.44.44	UP	8	SP-PE-R44.02
SP-P-R99	L2	Et0/0	10.99.88.99	UP	8	SP-P-R99.01

An interesting thing with IS-IS is: every device can see the other devices hostname, because there is a TLV for this feature and routers can include their hostnames inside that TLV. IS-IS is an extensible protocol thanks to these TLVs. Whenever new feature wants to be added to IS-IS, it's done using a new TLV. No need to Protocol rewrite!

```
SP-P-R88#show ip route isis | begin Gate
Gateway of last resort is not set

    10.0.0.0/8 is variably subnetted, 16 subnets, 2 masks
i L2    10.255.255.11/32 [115/10] via 10.88.11.11, 04:46:28, Ethernet0/1
i L2    10.255.255.22/32 [115/10] via 10.88.22.22, 00:03:49, Ethernet0/3
i L2    10.255.255.33/32 [115/10] via 10.88.33.33, 04:46:07, Ethernet0/2
i L2    10.255.255.44/32 [115/10] via 10.88.44.44, 00:03:49, Ethernet1/0
i L2    10.255.255.99/32 [115/10] via 10.99.88.99, 00:03:49, Ethernet0/0
```

Another interesting thing! Where are all /24 prefixes?! You remember we issued a command that caused IS-IS to only advertise passive /32 host NLRIs not inter-router connection.

As you can see, IS-IS's AD is 115, and all routes are L2.

```
SP-P-R88#show mpls ldp neighbor | include Peer
Peer LDP Ident: 10.255.255.99:0; Local LDP Ident 10.255.255.88:0
Peer LDP Ident: 10.255.255.33:0; Local LDP Ident 10.255.255.88:0
Peer LDP Ident: 10.255.255.44:0; Local LDP Ident 10.255.255.88:0
Peer LDP Ident: 10.255.255.11:0; Local LDP Ident 10.255.255.88:0
Peer LDP Ident: 10.255.255.22:0; Local LDP Ident 10.255.255.88:0
```

LDP neighborhood is also formed, and SP-P-R88 uses its Lo0 IPv4 address as LDP Identifier.

Let's analyze one neighbor in detail:

```
SP-P-R88#show mpls ldp neighbor
Peer LDP Ident: 10.255.255.99:0; Local LDP Ident 10.255.255.88:0
TCP connection: 10.255.255.99.37264 - 10.255.255.88.646
State: Oper; Msgs sent/rcvd: 438/431; Downstream
Up time: 06:04:51
LDP discovery sources:
Targeted Hello 10.255.255.88 -> 10.255.255.99, active, passive
Ethernet0/0, Src IP addr: 10.99.88.99
Addresses bound to peer LDP Ident:
10.255.255.99 10.99.22.99 10.99.44.99 10.99.11.99
10.99.33.99 10.99.88.99
```

TCP connection has been established for about 6 hours. We use Source port of 646 and the neighbor uses an ephemeral port of 37264.

Also targeted hellos are being generated using UDP between Loopback addresses.

```

> Frame 203: 76 bytes on wire (608 bits), 76 bytes captured (608 bits) on interface 0
> Ethernet II, Src: aa:bb:cc:00:20:00 (aa:bb:cc:00:20:00), Dst: aa:bb:cc:00:10:00 (aa:bb:cc:00:10:00)
> Internet Protocol Version 4, Src: 10.255.255.99, Dst: 10.255.255.88
> User Datagram Protocol, Src Port: 646, Dst Port: 646
  Label Distribution Protocol
    Version: 1
    PDU Length: 30
    LSR ID: 10.255.255.99
    Label Space ID: 0
  Hello Message
    0... .... = U bit: Unknown bit not set
    Message Type: Hello Message (0x100)
    Message Length: 20
    Message ID: 0x00000000
  > Common Hello Parameters
  IPv4 Transport Address
    00.. .... = TLV Unknown bits: Known TLV, do not Forward (0x0)
    TLV Type: IPv4 Transport Address (0x401)
    TLV Length: 4
    IPv4 Transport Address: 10.255.255.99
    
```

Take a look at Src and Dst Port numbers **they are both 646**.

Let's confirm that LDP has done its job and now we should have Labels for all /32 prefixes:

```

SP-PE-R11#show mpls fo | include 10.255.255
10000      80001  10.255.255.22/32 0          Et0/1      10.88.11.88
           90003  10.255.255.22/32 80         Et0/3      10.99.11.99
10001      Pop Label 10.255.255.99/32 0          Et0/3      10.99.11.99
10019      Pop Label 10.255.255.88/32 3246       Et0/1      10.88.11.88
10020      80004  10.255.255.33/32 0          Et0/1      10.88.11.88
           90001  10.255.255.33/32 0          Et0/3      10.99.11.99
10021      80000  10.255.255.44/32 0          Et0/1      10.88.11.88
           90004  10.255.255.44/32 0          Et0/3      10.99.11.99
    
```

Sounds good! You see ECMP and prefix labels from different neighbors for other PE's Lo0s. What about Pop Label ones? Where is the labels for those prefixes? This is because of something called PHP (not a programming language!) **Penultimate Hop Popping**. The default behavior of Cisco routers when they are the final destination for a particular prefix. The final destination of a prefix (The owner or the actual advertiser of the prefix) tells the neighboring router: Hey when you want to send me something that is destined for this prefix, please remove the outer label (yes the outer label because we can have multiple labels at the same time for a particular packet, this is called Label Stacking) and send it to me, I know what to do with that packet, I'm the owner of the prefix! But how the router tells the neighboring routers that do that way?

There is a label reserved for this purpose. Label number 3, this is called Implicit null. That Pop Label that we have highlighted stands for that. This router should pop the label for that destination before it sends it to the neighboring router.

Before we proceed with Next Task, Let me explain about P and PE routers in MPLS.

P stands for Provider device, and PE stands for Provider Edge device. We have another one called CE device. You can guess it stands for Customer Edge device.

When we say a device is P, it means that device only forwards packets based on Labels. To be exact: based on IGP created labels. This device is not going to talk to any Customer device and also we are not going to configure any BGP on this device. The Core devices on SP network are called P devices.

When we say a device is PE, it means that device can forward traffic that are labeled or unlabeled, and this device can be connected to Customer Edge devices. We run BGP on PE devices only. Have you heard of the most famous benefit of MPLS? Core Free BGP. We don't have to run BGP on Core (P) when we play with MPLS. The traffic between PE devices are going to be tunneled somehow that we will discuss soon!



Task 03:

- Configure iBGP on every PE router and **only** enable VPNv4 Unicast address family. All sessions should be established between Lo0 interfaces.
- The BGP AS number should be 10000.
- SP-PE-R11 and SP-PE-R22 should be configured as Route-Reflectors with the same Cluster-ID: 10.255.255.11.
- On Route Reflectors use session and policy templates to configure neighbor's session and policy parameters. Configure MD5 password "ccie_ei" for all sessions.



Solution:

In this Lab we don't have many PE devices, we can configure full mesh peering between them, but in real life situations you face many PE devices. For example more than PE devices you see in an SP network. Can you do a full mesh between all of them? It would be a nightmare! I almost forgot to mention why we even need Route-Reflectors! In SP network an even in MPLS Enabled campus network all Edge Devices form iBGP peering together. BGP's Loop Prevention mechanism assists AS-Path Attribute in order to avoid loops. When an eBGP peer receives NLRI's that are bounded to some attributes, it looks at AS-Path attribute, if the peer sees itself AS number in the AS-Path List, it ignores that reachability information.

When we form iBGP peering, there is no loop prevention mechanism like AS-Path because a router only prepends it's AS# to the Reachability Information when it sends them to an eBGP neighbor (a neighbor with different AS#). So it is dangerous to advertise an iBGP learned route to other iBGP peer and we should configure full mesh peering between all iBGP peers.

As I already mentioned it is not a scalable method. We can use Confederations or Route Reflectors instead.

A Route Reflector (RR) can get a route from an iBGP client and advertise it to another iBGP client. I don't want to get into all RR capabilities to confuse you. All you need to know for our Lab is an RR gets a route from a client and advertises it to another client.

How does an RR prevent loops? A good question! RR uses a field called Cluster-ID, when a route receives from a client, RR puts it's cluster-id into that route and sends it to another neighbor, by looking at cluster-id field routers can know if that route it advertised back to them.


```
SP-PE-R33#show bgp vpnv4 uni all 172.19.255.4/32
!
 10.255.255.22 (metric 20) (via default) from 10.255.255.11 (10.255.255.11)
   Origin incomplete, metric 11, localpref 100, valid, internal
   Extended Community: RT:10000:4 OSPF DOMAIN ID:0x0005:0x000000065004
     OSPF RT:0.0.0.0:2:0 OSPF ROUTER ID:172.19.255.22:0
   Originator: 10.255.255.22, Cluster list: 10.255.255.11
   mpls labels in/out nolabel/20013
   rx pathid: 0, tx pathid: 0
```

Take a look at **Cluster List** and **Originator field**. Every RR adds its cluster ID to that list. Then by assisting that list they can prevent loops. The Originator is the owner of the route, in this example it is another Client of our RR.

Let's configure our first Route-Reflector:

```
SP-PE-R11:
router bgp 10000
 template peer-policy RR-CLIENTS-TEMPLATE
   route-reflector-client
 exit-peer-policy
!
 template peer-session iBGP-SESSION-TEMPLATE
   remote-as 10000
   password ccie_ei
   update-source Loopback0
 exit-peer-session
!
bgp cluster-id 10.255.255.11
bgp log-neighbor-changes
no bgp default ipv4-unicast
neighbor 10.255.255.22 inherit peer-session iBGP-SESSION-TEMPLATE
neighbor 10.255.255.33 inherit peer-session iBGP-SESSION-TEMPLATE
neighbor 10.255.255.44 inherit peer-session iBGP-SESSION-TEMPLATE
!
address-family ipv4
 exit-address-family
```

```
!  
address-family vpnv4  
neighbor 10.255.255.22 activate  
neighbor 10.255.255.22 send-community extended  
neighbor 10.255.255.33 activate  
neighbor 10.255.255.33 send-community extended  
neighbor 10.255.255.33 inherit peer-policy RR-CLIENTS-TEMPLATE  
neighbor 10.255.255.44 activate  
neighbor 10.255.255.44 send-community extended  
neighbor 10.255.255.44 inherit peer-policy RR-CLIENTS-TEMPLATE  
exit-address-family  
!
```

First we define a BGP process with AS# 10000, by default IOS/IOS-XE enables IPv4 Unicast address family for every peer that we define. But the task explicitly have asked us to enable only VPNv4 Unicast Address Family. To disable this default behavior we need to enter **no bgp default ipv4-unicast command**. Now we must activate whatever AF we want to be enabled for a particular neighbor.

By default BGP router-id the same steps that it uses for OSPF. And also sets router-id as cluster-id. The task explicitly have asked us to configure it manually the same value on both RR. That's why we must enter **bgp cluster-id 10.255.255.11** command.

Then we define Templates. Session template can contain any parameter that is needed for BGP session formation:

```
SP-PE-R11(config-router)# template peer-session iBGP-SESSION-TEMPLATE  
SP-PE-R11(config-router-stmp)#?  
BGP peer-policy configuration commands:  
bmp-activate          Activate the BMP monitoring for a BGP peer  
cluster-id            Configure Route-Reflector Cluster-id (peers may  
                      reset)  
default               Set a command to its defaults  
description           Neighbor specific description  
disable-connected-check one-hop away EBGP peer using loopback address  
ebgp-multihop         Allow EBGP neighbors not on directly connected  
                      networks  
exit-peer-session     Exit from template configuration mode
```

fall-over	session fall on peer route lost
ha-mode	high availability mode
inherit	Inherit a template
local-as	Specify a local-as number
no	Negate a command or set its defaults
password	Set a password
path-attribute	BGP optional attribute filtering
remote-as	Specify a BGP neighbor
shutdown	Administratively shut down this neighbor
timers	BGP per neighbor timers
transport	Transport options
ttl-security	BGP ttl security check
update-source	Source of routing updates
version	Set the BGP version to match a neighbor

Take a look at inherit word in the neighbor commands. All those neighbors inherits attributes we defined under the templates. And also activate word in the neighbor command, this command enables that particular AF for a neighbor.

Cisco IOS/IOS-XE automatically adds send-community extended command after you activate VPNv4 for a neighbor. No need to enter it manually.

VPNv4 uses extended communities a lot! Then it is necessary for this VPN service.

Let's configure the other RR:

```
SP-PE-R22:
router bgp 10000
  template peer-policy RR-CLIENTS-TEMPLATE
  route-reflector-client
exit-peer-policy
!
template peer-session iBGP-SESSION-TEMPLATE
  remote-as 10000
  password ccie_ei
  update-source Loopback0
exit-peer-session
!
```

```

bgp cluster-id 10.255.255.11
bgp log-neighbor-changes
no bgp default ipv4-unicast

neighbor 10.255.255.11 inherit peer-session iBGP-SESSION-TEMPLATE
neighbor 10.255.255.33 inherit peer-session iBGP-SESSION-TEMPLATE
neighbor 10.255.255.44 inherit peer-session iBGP-SESSION-TEMPLATE
!
address-family ipv4
exit-address-family
!
address-family vpnv4
  neighbor 10.255.255.11 activate
  neighbor 10.255.255.11 send-community extended
  neighbor 10.255.255.33 activate
  neighbor 10.255.255.33 send-community extended
  neighbor 10.255.255.33 inherit peer-policy RR-CLIENTS-TEMPLATE
  neighbor 10.255.255.44 activate
  neighbor 10.255.255.44 send-community extended
  neighbor 10.255.255.44 inherit peer-policy RR-CLIENTS-TEMPLATE
exit-address-family
!
  
```

The important thing is: RR's must be fully meshed, in this scenario they are neighbors with each other and there is no route-reflector-client because they are not client of each other.

It's time to configure clients:

SP-PE-R33 and SP-PE-R44:

```

router bgp 10000
  bgp log-neighbor-changes
  no bgp default ipv4-unicast
  neighbor 10.255.255.11 remote-as 10000
  neighbor 10.255.255.11 password ccie_ei
  neighbor 10.255.255.11 update-source Loopback0
  neighbor 10.255.255.22 remote-as 10000
  neighbor 10.255.255.22 password ccie_ei
  neighbor 10.255.255.22 update-source Loopback0
  
```

```
address-family vpnv4
  neighbor 10.255.255.11 activate
  neighbor 10.255.255.11 send-community extended
  neighbor 10.255.255.22 activate
  neighbor 10.255.255.22 send-community extended
exit-address-family
!
```

Just copy paste the same configurations on all client PEs of this cluster!

Congratulations! Our SP network is ready to provide L3VPN services to customers.

Take a rest and continue with the next Task!



Task 04:

- Provide MPLS L3VPN Service to CustA (Customer A)
- Use Static-Routes as PE-CE Routing Protocol
- CustA-Static-CE1 device should reach other sites using it's Lo0 interface and also reach the router that is doing NAT (CustA-2-Static-CE12 device) using it's Lo200 interface.



Solution:

First let's talk about MPLS L3VPN service. Using MPLS VPNs an SP can provide both L2 and L3 connectivity between customer sites. L2 Connectivity means, customer sites can reach together just like they are connected to an L2 switch, they can reach each other using a subnet for example 172.16.10.0/24, each site router's (CE device) interface would have an IP address in that subnet, in other words broadcast, unknown unicast and multicast traffic (BUM Traffic) will be flooded to every site using services like VPLS (Virtual Private LAN Service, a P2MP connection).

It sounds good to have this kind of connectivity, but the problem is Scalability. What if a customer has many sites like a 1000 or more? Or customer wants redundant connections on many sites? Using VPLS it's a nightmare to have this kind of connections, only a single misconfiguration or a misbehaving device would cause a loop in the network and bring down the whole network!

L3VPN service comes into play, a Customer on each of its sites, gives the Service Provider its reachability information (the routing table information) then Service Provider routers somehow spread this reachability information to other sites.

We discussed about CE and PE devices, and now we know the Customer side device is called CE (Customer Edge) and the Provider side device is called PE (Provider Edge). In L3VPN service CE must tell its reachability information to PE device and also CE must get other sites reachability information from the PE device.

There are different ways of doing this job. We can use different routing protocols such as Static one (Static-Routes), or Dynamic ones like BGP, OSPF, EIGRP, RIP or even IS-IS (but it's not recommended! I will tell you why in this workbook).

Let's continue with the task.

First configure Site 2 and 3 of the CustA. Every CE device has a Loopback IP address. On CustA-Static-CE2 we have Lo0: 172.16.255.2/32 and also a subnet connected to the SP-PE-R22 172.16.22.0/24, on the other site for CustA-Static-CE3 we have Lo0: 172.16.255.3/32 and subnet connected to SP-PE-R33 172.16.33.0/24.

What is common on these subnets? 172.16.0.0/16 right? Then we only need a single Static route on every CE device pointing to next-hop of the appropriate PE device:

CustA-Static-CE2:

```
interface Loopback0
 ip address 172.16.255.2 255.255.255.255
!
interface Ethernet1/0
 ip address 172.16.22.2 255.255.255.0
!
ip route 172.16.0.0 255.255.0.0 172.16.22.22
```

We only configured IP addresses on interfaces then wrote a static-route for destination 172.16.0.0/16 to the next-hop IP address of the PE interface: 172.16.22.22.

The same for CustA-Static-CE3:

CustA-Static-CE3:

```
interface Loopback0
 ip address 172.16.255.3 255.255.255.255
!
interface Ethernet1/2
 ip address 172.16.33.3 255.255.255.0
!
ip route 172.16.0.0 255.255.0.0 172.16.33.33
```

Now time to configure PE devices:

A Service Provider can have many customers with overlapping IP addresses. In this scenario I have not used overlapping IP addresses for different customers, every customer uses different IPv4 subnets. But it is only a lab environment, an SP cannot force customers to follow their IP addressing scheme! It's possible for example 100 customers use 192.168.1.0/24.

A Service Provider must somehow isolate customer traffic, then no one can see each other's traffic. You are already familiar with VRF-Lite/VRF and you know that by using VRF-Lite we can have Isolated instances of routing table at the same time. Without using VRF-Lite we can only have 1 routing table on a router, it's called Global Routing Table. We can create many routing and forwarding instances (the number differs from platform to platform) using a mechanism called VRF. If you don't use VRF along with MP-BGP for MPLS L3VPN then we call it VRF-Lite.

Now we create a VRF for than customer on PEs:

SP-PE-R22:

```
vrf definition CustA-Static
rd 10000:1
!
address-family ipv4
  route-target export 10000:1
  route-target import 10000:1
exit-address-family
!
```

By using **vrf definition** command we define a VRF instance named CustA-Static, you could also use **ip vrf CustA-Static** command for this scenario because we don't have any IPv6 addresses here (You got the point, **vrf definition** command both supports IPv4 and IPv6 address families).

So what is RD (Route-Distinguisher) here? You remember we configured an MP-BGP AF before, named VPNv4 Unicast. BGP is going to do the magic for us in L3VPNs! The routes that are learned from each CE device are going to be distributed to other CE devices on another PEs using MP-BGP AF VPNv4 Unicast. We discussed about that each customer reachability information should be isolated from other customers, so we need to add something to their routes in order to make them unique between other customers. The RD (Route-Distinguisher) is going to do exactly what we need to do, adding a unique 32 bit in the beginning of an NLRI makes it unique (it is actually 64 bit, but 32 bit is considered by all implementations). Then we have 32-bit IPv4 prefix + 32-bit prefix-length + 32-bit RD = 96-bit unique NLRI.

So if 2 customers use the same prefix/prefix length with different RDs, the result is like this:

Cust1 RD=10000:1, Cust2 RD=10000:2

Cust1 NLRI: 10000:1:192.168.1.0/24

Cust2 NLRI: 10000:2:192.168.1.0/24

Different routes even if they are overlapping right?

```
SP-PE-R22#show bgp vpnv4 uni rd 10000:1 172.16.255.1/32
BGP routing table entry for 10000:1:172.16.255.1/32, version 58
Paths: (1 available, best #1, table CustA-Static)
  Advertised to update-groups:
    2
!
!
```


What about RT (Route-Target)? When a PE gets routes from a CE device, it places them into the VRF routing table, we need a mechanism to export these routes from the VRF table and send them to other PE devices, we must redistribute these routes into MP-BGP and also tag them to let the other PEs to import them selectively, because they don't care about the RD for doing this job, they only import what we explicitly tell them to import.

On SP-PE-R22 we told the PE under VRF CustA-Static to export routes that have received from the CE (the routes that has in its vrf routing table now) using an extended community (let's name it tag) of 10000:1 and send it to other PEs. And also we told the PE that import routes that have extended community tag of 10000:1.

We could use different RD and RT for a customer. To make this scenario simple and understandable we used the same 10000:1 for both RD/RT.

But there is an important rule: we can only define one RD for a VRF on a PE but we can have multiple RTs for that VRF on a PE.

Now let's configure PE interfaces and static routes:

SP-PE-R22:

```
interface Ethernet1/0
  vrf forwarding CustA-Static
  ip address 172.16.22.22 255.255.255.0
!
ip route vrf CustA-Static 172.16.255.2 255.255.255.255 Ethernet1/0 172.16.22.2
```

Take a look at the **vrf forwarding** command, this command makes interface e1/0 to be part of vrf CustA-Static and we should write a static route in order to reach CustA-Static-CE2 Loopback 0 interface, this route also needs to be part of that VRF instance because of that we include **vrf** after **ip route** command.

The last thing we need to do is to redistribute that route into MP-BGP VPNv4 unicast AF:

SP-PE-R22:

```
router bgp 10000
!
address-family ipv4 vrf CustA-Static
  redistribute connected
  redistribute static
exit-address-family
!
```

Under BGP process we should define an IPv4 AF for that VRF instance and redistribute static routes into it. It is not necessary for other sites to also reach the interface between PE-CE (connected/direct network) but we do so, let's configure full reachability between sites even PE-CE connections.

It's time to configure the other site:

```
SP-PE-R33:
vrf definition CustA-Static
  rd 10000:1
  !
  address-family ipv4
    route-target export 10000:1
    route-target import 10000:1
  exit-address-family
!
!
interface Ethernet1/2
  vrf forwarding CustA-Static
  ip address 172.16.33.33 255.255.255.0
!
ip route vrf CustA-Static 172.16.255.3 255.255.255.255 Ethernet1/2 172.16.33.3
router bgp 10000
!
  address-family ipv4 vrf CustA-Static
    redistribute connected
    redistribute static
  exit-address-family
!
```



Verification:

```
SP-PE-R22#show ip route vrf CustA-Static | begin Gateway
```

```
Gateway of last resort is not set
```

```
172.16.0.0/16 is variably subnetted, 5 subnets, 2 masks
```

```

C      172.16.22.0/24 is directly connected, Ethernet1/0
L      172.16.22.22/32 is directly connected, Ethernet1/0
B      172.16.33.0/24 [200/0] via 10.255.255.33, 03:05:57
S      172.16.255.2/32 [1/0] via 172.16.22.2, Ethernet1/0
B      172.16.255.3/32 [200/0] via 10.255.255.33, 03:05:57
    
```

```

SP-PE-R33#show ip route vrf CustA-Static | begin Gateway
    
```

```

Gateway of last resort is not set
    
```

```

172.16.0.0/16 is variably subnetted, 5 subnets, 2 masks
    
```

```

B      172.16.22.0/24 [200/0] via 10.255.255.22, 03:06:43
C      172.16.33.0/24 is directly connected, Ethernet1/2
L      172.16.33.33/32 is directly connected, Ethernet1/2
B      172.16.255.2/32 [200/1] via 10.255.255.22, 03:06:43
S      172.16.255.3/32 [1/0] via 172.16.33.3, Ethernet1/2
    
```

Look at vrf word after **show ip route** command and also B (BGP learned routes) and 200 AD because those are iBGP routes (Every PE forms iBGP session with other PEs, the same AS#).

To view MP-BGP learned routes on BGP use command **show bgp vpnv4 unicast rd 10000:1**

```

SP-PE-R22#show bgp vpnv4 uni rd 10000:1 | begin Network
    
```

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 10000:1 (default for vrf CustA-Static)					
*> 172.16.22.0/24	0.0.0.0	0		32768	?
*>i 172.16.33.0/24	10.255.255.33	0	100	0	?
*> 172.16.255.2/32	172.16.22.2	1		32768	?
*>i 172.16.255.3/32	10.255.255.33	0	100	0	?

```

SP-PE-R33#show bgp vpnv4 uni rd 10000:1 | begin Network
    
```

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 10000:1 (default for vrf CustA-Static)					
* i 172.16.22.0/24	10.255.255.22	0	100	0	?
*>i	10.255.255.22	0	100	0	?
*> 172.16.33.0/24	0.0.0.0	0		32768	?
* i 172.16.255.2/32	10.255.255.22	1	100	0	?
*>i	10.255.255.22	1	100	0	?

SP-PE-R33 has two routes for other PE's networks, because it gets them from two redundant Route-Reflectors.

To get detailed information about each route we can use the prefix at the end of previous command:

```
SP-PE-R22#show bgp vpnv4 uni rd 10000:1 172.16.255.3/32
BGP routing table entry for 10000:1:172.16.255.3/32, version 29
Paths: (1 available, best #1, table CustA-Static)
  Advertised to update-groups:
    1          2
  Refresh Epoch 1
  Local, (Received from a RR-client)
    10.255.255.33 (metric 20) (via default) from 10.255.255.33 (10.255.255.33)
    Origin incomplete, metric 0, localpref 100, valid, internal, best
    Extended Community: RT:10000:1
    mpls labels in/out no-label/30006
    rx pathid: 0, tx pathid: 0x0
```

You see the label number **30006** for this route? This is the label that PE33 told PE22 using MP-BGP VPNv4 unicast AF: "If you want to reach this network include this label at the bottom of all labels". If we traceroute on CE2 we should see this label:

```
CustA-Static-CE2#trace 172.16.255.3 source lo0 probe 1
Type escape sequence to abort.
Tracing the route to 172.16.255.3
VRF info: (vrf in name/id, vrf out name/id)
 1 172.16.22.22 1 msec
 2 10.88.22.88 [MPLS: Labels 80002/30006 Exp 0] 2 msec
 3 172.16.33.33 [MPLS: Label 30006 Exp 0] 1 msec
 4 172.16.33.3 2 msec
```

Interesting! What about the other Label? 80002:

The label number 80002 is the label to reach the other PE device (SP-PE-R33).

We can have something called Label Stacking in MPLS, mean multiple labels for a packet:

```
SP-PE-R22#show mpls forwarding-table
```

Local Label	Outgoing Label	Prefix or Tunnel Id	Bytes Switched	Label	Outgoing interface	Next Hop
20000	Pop Label	10.255.255.88/32	0		Et0/3	10.88.22.88
20001	Pop Label	10.255.255.99/32	0		Et0/1	10.99.22.99
20002	80000	10.255.255.11/32	0		Et0/3	10.88.22.88
	90001	10.255.255.11/32	0		Et0/1	10.99.22.99
20003	80002	10.255.255.33/32	0		Et0/3	10.88.22.88
	90003	10.255.255.33/32	0		Et0/1	10.99.22.99
20004	80003	10.255.255.44/32	0		Et0/3	10.88.22.88
	90004	10.255.255.44/32	0		Et0/1	10.99.22.99

The label number 80002 is the label to reach the other PE device (SP-PE-R33).

We can have something called Label Stacking in MPLS, mean multiple labels for a packet:

```
> Frame 13: 122 bytes on wire (976 bits), 122 bytes captured (976 bits) on interface 0
> Ethernet II, Src: aa:bb:cc:00:60:30 (aa:bb:cc:00:60:30), Dst: aa:bb:cc:00:10:30 (aa:bb:cc:00:10:30)
> MultiProtocol Label Switching Header, Label: 80002, Exp: 0, S: 0, TTL: 254
> MultiProtocol Label Switching Header, Label: 30006, Exp: 0, S: 1, TTL: 254
  0000 0111 0101 0011 0110 ..... = MPLS Label: 30006
  ..... = MPLS Experimental Bits: 0
  .....1 ..... = MPLS Bottom Of Label Stack: 1
  ..... 1111 1110 = MPLS TTL: 254
> Internet Protocol Version 4, Src: 172.16.255.2, Dst: 172.16.255.3
> Internet Control Message Protocol
```

Interesting right? A packet with multiple Labels, this is the power of MPLS and its capability to tunnel the traffic between two end points. Core Routers are not aware of the bottom label they only look at the top label 80002:

```
SP-P-R88#show mpls forwarding-table
```

Local Label	Outgoing Label	Prefix or Tunnel Id	Bytes Switched	Label	Outgoing interface	Next Hop
80000	Pop Label	10.255.255.11/32	1704		Et0/1	10.88.11.11
80001	Pop Label	10.255.255.22/32	45577		Et0/3	10.88.22.22
80002	Pop Label	10.255.255.33/32	43734		Et0/2	10.88.33.33
80003	Pop Label	10.255.255.44/32	1648		Et1/0	10.88.44.44
80004	Pop Label	10.255.255.99/32	76		Et0/0	10.99.88.99

The Core router LFIB proves that it has no information about bottom label.

Now you realize why these services are called VPN. They are Virtual and Private and also the traffic is being tunneled.

Time to configure Site 1:

The task has asked us to use something else than just doing normal VRF, CustA-Static-CE1 needs to reach the other sites (Site 2 and 3) using its Loopback 0 interface and also reach another Site (NAT doing Router) using its Loopback 200 interface.

To do so, we need to do something like PBR, Selectively put the traffic to the right VRF.

Configure an access list and match those traffic based on source IP address:

SP-PE-R11:

```
ip access-list standard CustA-2-Lo200
 permit 10.200.255.1

ip access-list standard CustA-Lo0
 permit 172.16.255.1
```

Create 2 different VRFs:

SP-PE-R11:

```
vrf definition CustA-Static
 rd 10000:1
 !
 address-family ipv4
  route-target export 10000:1
  route-target import 10000:1
 exit-address-family
 !
```

SP-PE-R11:

```
vrf definition CustA-2-Static
 rd 10000:1111
 !
 address-family ipv4
  route-target export 10000:1111
  route-target import 10000:1111
 exit-address-family
 !
```

And then define a route-map to put the traffic with different source IP addresses to different VRFs:

SP-PE-R11:

```
route-map CustA-VRF-Receive-MAP permit 10
  match ip address CustA-2-Lo200
  set vrf CustA-2-Static
route-map CustA-VRF-Receive-MAP permit 20
  match ip address CustA-Lo0
  set vrf CustA-Static
```

Look at the CustA-Static-CE1 facing interface configuration:

SP-PE-R11:

```
interface Ethernet0/2
  ip policy route-map CustA-VRF-Receive-MAP
  ip vrf receive CustA-Static
  ip vrf receive CustA-2-Static
  ip address 172.16.11.11 255.255.255.0
```

Instead of putting the interface under a VRF (using **vrf forwarding** command), we use **ip vrf receive** [vrf name] command and also use the route-map that we have defined previously by entering **ip policy route-map** command, Just like PBR.

Configure static routes for 2 VRFs in order to reach Lo0/Lo200 interface of CE1:

SP-PE-R11:

```
ip route vrf CustA-2-Static 10.200.255.1 255.255.255.255 Ethernet0/2 172.16.11.1
ip route vrf CustA-Static 172.16.255.1 255.255.255.255 Ethernet0/2 172.16.11.1
```

One more step (Redistribution into MP-BGP):

SP-PE-R11:

```
router bgp 10000
!
address-family ipv4 vrf CustA-Static
  redistribute static
exit-address-family
!
address-family ipv4 vrf CustA-2-Static
  redistribute static
exit-address-family
!
```

CustA-Static-CE1:

```
interface Loopback0
 ip address 172.16.255.1 255.255.255.255
!
interface Loopback200
 ip address 10.200.255.1 255.255.255.255
!
interface Ethernet0/2
 ip address 172.16.11.1 255.255.255.0
!
ip route 172.16.0.0 255.255.0.0 172.16.11.11
ip route 200.0.0.0 255.0.0.0 172.16.11.11
```

We need static routes in order to reach other sites and also outside network 200.0.0.0/8 (NAT doing router connected).

Now CE1 can ping CE2 and CE3 using Loopback0 IP addresses:

```
CustA-Static-CE1#ping 172.16.255.2 source lo 0 re 10
Type escape sequence to abort.
Sending 10, 100-byte ICMP Echos to 172.16.255.2, timeout is 2 seconds:
Packet sent with a source address of 172.16.255.1
!!!!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 1/1/2 ms

CustA-Static-CE1#ping 172.16.255.3 source lo 0 re 10
Type escape sequence to abort.
Sending 10, 100-byte ICMP Echos to 172.16.255.3, timeout is 2 seconds:
Packet sent with a source address of 172.16.255.1
!!!!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 1/1/2 ms
```

What about packets with Loopback 200 source? We have not configured Site 12 yet (NAT doing router site):

SP-PE-R33:

```
vrf definition CustA-2-Static
 rd 10000:1111
 !
 address-family ipv4
  route-target export 10000:1111
  route-target import 10000:1111
 exit-address-family
 !
 !
 interface Ethernet0/1
  vrf forwarding CustA-2-Static
  ip address 10.16.133.33 255.255.255.0
 !
 router bgp 10000
 !
 address-family ipv4 vrf CustA-2-Static
  redistribute static
 exit-address-family
 !
 ip route vrf CustA-2-Static 200.0.0.0 255.0.0.0 10.16.133.12
```

It is also necessary for SP-PE-R33 to know how to reach Network 200.0.0.0/8.

One more step to complete the Task:

EX13:

```
interface Loopback0
 ip address 200.13.13.13 255.255.255.255
 !
 interface Ethernet0/0
 ip address 200.12.13.13 255.255.255.0
 !
```

CustA-2-Static-CE12:

```

interface Ethernet0/0
 ip address 200.12.13.12 255.255.255.0
 ip nat outside
 ip virtual-reassembly in
!
interface Ethernet0/1
 ip address 10.16.133.12 255.255.255.0
 ip nat inside
 ip virtual-reassembly in
!
ip nat inside source list Local-Range interface Ethernet0/0 overload
ip route 10.200.255.1 255.255.255.255 10.16.133.33
ip route 200.0.0.0 255.0.0.0 200.12.13.13
ip access-list standard Local-Range
 permit 10.200.255.1
    
```

Well done! Time to do some verifications.


Verification:

```

CustA-Static-CE1#ping 200.13.13.13 source lo 200 re 10
Type escape sequence to abort.
Sending 10, 100-byte ICMP Echos to 200.13.13.13, timeout is 2 seconds:
Packet sent with a source address of 10.200.255.1
!!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 1/2/3 ms

CustA-Static-CE12#show ip nat translations
Pro Inside global      Inside local      Outside local      Outside global
icmp 200.12.13.12:4    10.200.255.1:4    200.13.13.13:4    200.13.13.13:4

SP-PE-R11#show bgp vpnv4 uni rd 10000:1 | begi Net
      Network      Next Hop      Metric LocPrf Weight Path
Route Distinguisher: 10000:1 (default for vrf CustA-Static)
    
```

```
*>i 172.16.22.0/24 10.255.255.22 0 100 0 ?
*>i 172.16.33.0/24 10.255.255.33 0 100 0 ?
*> 172.16.255.1/32 172.16.11.1 0 32768 ?
*>i 172.16.255.2/32 10.255.255.22 1 100 0 ?
*>i 172.16.255.3/32 10.255.255.33 0 100 0 ?
```

```
SP-PE-R11#show bgp vpnv4 uni rd 10000:1111 | begin Net
```

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 10000:1111 (default for vrf CustA-2-Static)					
*>i 10.16.133.0/24	10.255.255.33	0	100	0	?
*> 10.200.255.1/32	172.16.11.1	0		32768	?
*>i 200.0.0.0/8	10.255.255.33	0	100	0	?

```
CustA-Static-CE12#clear ip nat translation *
```

```
CustA-Static-CE1#trace 200.13.13.13 source lo 200 probe 5
```

```
Type escape sequence to abort.
```

```
Tracing the route to 200.13.13.13
```

```
VRF info: (vrf in name/id, vrf out name/id)
```

```
 1 172.16.11.11 1 msec 0 msec 1 msec 1 msec 0 msec
 2 10.88.11.88 [MPLS: Labels 80002/30020 Exp 0] 2 msec 2 msec 1 msec 1 msec 1 msec
 3 10.16.133.33 [MPLS: Label 30020 Exp 0] 2 msec 1 msec 1 msec 2 msec 1 msec
 4 10.16.133.12 2 msec 1 msec 2 msec 1 msec 1 msec
 5 200.12.13.13 2 msec * 2 msec * 2 msec
```

```
CustA-Static-CE12#show ip nat translations
```

Pro	Inside global	Inside local	Outside local	Outside global
udp	200.12.13.12:49208	10.200.255.1:49208	200.13.13.13:33454	200.13.13.13:33454
udp	200.12.13.12:49209	10.200.255.1:49209	200.13.13.13:33455	200.13.13.13:33455
udp	200.12.13.12:49210	10.200.255.1:49210	200.13.13.13:33456	200.13.13.13:33456
udp	200.12.13.12:49211	10.200.255.1:49211	200.13.13.13:33457	200.13.13.13:33457
udp	200.12.13.12:49212	10.200.255.1:49212	200.13.13.13:33458	200.13.13.13:33458



Task 05:

- Provide MPLS L3VPN Service to CustB (Customer B)
- Use RIP as PE-CE Routing Protocol



Solution:

We have already done L3VPN service using Static-Routes. It's not a scalable solution, for scalability we have to use Dynamic Routing Protocols like BGP, OSPF, RIP, EIGRP and even IS-IS. Each of them has advantages and disadvantages.

The steps to configure L3VPN using RIP as a PE-CE Routing Protocol are the same as Previous tasks (Static-Routes) plus one more step.

First Configure RIP on CE routers:

CustB-RIP-CE1:

```
interface Loopback0
  ip address 172.17.255.1 255.255.255.255
!
interface Ethernet0/0
  ip address 172.17.11.1 255.255.255.0
!
router rip
  version 2
  passive-interface default
  no passive-interface Ethernet0/0
  network 172.17.0.0
  no auto-summary
!
```

CustB-RIP-CE2:

```
interface Loopback0
  ip address 172.17.255.2 255.255.255.255
!
interface Ethernet0/0
  ip address 172.17.22.2 255.255.255.0
!
router rip
```

```
version 2
passive-interface default
no passive-interface Ethernet0/0
network 172.17.0.0
no auto-summary
!
CustB-RIP-CE3:
interface Loopback0
 ip address 172.17.255.3 255.255.255.255
!
interface Ethernet1/0
 ip address 172.17.44.3 255.255.255.0
!
router rip
 version 2
 passive-interface default
 no passive-interface Ethernet1/0
 network 172.17.0.0
 no auto-summary
!
```

Just an easy RIP configuration on Customer Sites.

PE Configurations:

```
SP-PE-R11:
vrf definition CustB-RIP
 rd 10000:2
!
 address-family ipv4
  route-target export 10.255.255.11:2
  route-target import 10.255.255.22:2
  route-target import 10.255.255.44:2
 exit-address-family
!
!
interface Ethernet0/0
```

```

vrf forwarding CustB-RIP
 ip address 172.17.11.11 255.255.255.0
 !
router bgp 10000
 !
 address-family ipv4 vrf CustB-RIP
  redistribute rip
 exit-address-family
 !
router rip
 !
 address-family ipv4 vrf CustB-RIP
  redistribute bgp 10000
  network 172.17.0.0
  default-metric 2
  no auto-summary
  version 2
 exit-address-family
 !

```

Look at Highlighted commands, Instead of using the same Route-Target values as RD, this time I used different values for import and export.

Lo0 interface address of PE router + Customer unique number:

`route-target export 10.255.255.11:2` command causes PE 11 to export customer routes from VRF table and tag them using RT Extended Community value of 10.255.255.11:2 then send them to other PEs using VPNv4 Unicast MP-BGP Address Family.

Other PE routers must be explicitly configured for this value in order to import these routes into the appropriate VRF routing table.

You can see on PE 11 we also configured

```
route-target import 10.255.255.22:2 and route-target import 10.255.255.44:2
```

using these two commands this router is going to import the routes that PE 22 and PE 44 have exported from CustB-RIP VRF routing table.

What is the benefit of doing this way instead of using the same value for both import/export RT values? The reason is Flexibility and Inter-VRF route leaking!

We can selectively import/export whatever site routes we want. If a Customer does not want to have connectivity between 2 of their sites but the other sites stay fully connected, SP can do it for the customer in an easy way using RT import/export values.

The other benefit is route leaking between 2 or many customers. I will show you how it works.

Look at this configuration:

CustB-RIP-CE1:

```
router rip
!
address-family ipv4 vrf CustB-RIP
```

Many thanks to Cisco! They have made RIP and other Routing protocols like OSPF, EIGRP to be VRF aware. Using this VRF awareness RIP can send/receive routes to/from Customer CE device.

An important step is mutual redistribution between RIP/BGP. Routes have received from CE device must be redistributed into MP-BGP and also routes received from MP-BGP (Other PEs) need to be redistributed into RIP.

SP-PE-R11:

```
router bgp 10000
!
address-family ipv4 vrf CustB-RIP
 redistribute rip
 exit-address-family
!
router rip
!
address-family ipv4 vrf CustB-RIP
 redistribute bgp 10000
 default-metric 2
 exit-address-family
!
```

Seed metric is also necessary for RIP, Because by default RIP does not have any seed metric and routes are not going to be redistributed from BGP. Another way of setting metric is during redistribution: redistribute bgp 10000 metric 1

Configure other PEs the same way:

SP-PE-R22:

```
vrf definition CustB-RIP
  rd 10000:2
  !
  address-family ipv4
    route-target export 10.255.255.22:2
    route-target import 10.255.255.11:2
    route-target import 10.255.255.44:2
  exit-address-family
!
!
interface Ethernet0/0
  vrf forwarding CustB-RIP
  ip address 172.17.22.22 255.255.255.0
!
router bgp 10000
  !
  address-family ipv4 vrf CustB-RIP
    redistribute rip
  exit-address-family
!
router rip
  !
  address-family ipv4 vrf CustB-RIP
    redistribute bgp 10000 metric 1
    network 172.17.0.0
    no auto-summary
    version 2
  exit-address-family
!
```

SP-PE-R44:

```
vrf definition CustB-RIP
  rd 10000:2
  !
```



```

address-family ipv4
  route-target export 10.255.255.44:2
  route-target import 10.255.255.11:2
  route-target import 10.255.255.22:2
exit-address-family
!
!
interface Ethernet0/1
  vrf forwarding CustB-RIP
  ip address 172.17.44.44 255.255.255.0
!
router bgp 10000
!
  address-family ipv4 vrf CustB-RIP
  redistribute rip
  exit-address-family
!
router rip
!
  address-family ipv4 vrf CustB-RIP
  redistribute bgp 10000 metric 1
  network 172.17.0.0
  no auto-summary
  version 2
  exit-address-family
!
    
```



Verification:

```
SP-PE-R11#show bgp vpnv4 unicast rd 10000:2 | begin Net
```

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 10000:2 (default for vrf CustB-RIP)					
*> 172.17.11.0/24	0.0.0.0	0		32768	?
*>i 172.17.22.0/24	10.255.255.22	0	100	0	?
*>i 172.17.44.0/24	10.255.255.44	0	100	0	?
*> 172.17.255.1/32	172.17.11.1	1		32768	?
*>i 172.17.255.2/32	10.255.255.22	1	100	0	?
*>i 172.17.255.3/32	10.255.255.44	1	100	0	?

```
SP-PE-R22#show bgp vpnv4 unicast rd 10000:2 | begin Net
```

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 10000:2 (default for vrf CustB-RIP)					
*>i 172.17.11.0/24	10.255.255.11	0	100	0	?
*> 172.17.22.0/24	0.0.0.0	0		32768	?

```
*>i 172.17.44.0/24 10.255.255.44 0 100 0 ?
*>i 172.17.255.1/32 10.255.255.11 1 100 0 ?
*> 172.17.255.2/32 172.17.22.2 1 32768 ?
*>i 172.17.255.3/32 10.255.255.44 1 100 0 ?
```

```
SP-PE-R44#show bgp vpnv4 unicast rd 10000:2 | begin Net
```

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 10000:2 (default for vrf CustB-RIP)					
* i 172.17.11.0/24	10.255.255.11	0	100	0	?
*>i	10.255.255.11	0	100	0	?
* i 172.17.22.0/24	10.255.255.22	0	100	0	?
*>i	10.255.255.22	0	100	0	?
*> 172.17.44.0/24	0.0.0.0	0		32768	?
* i 172.17.255.1/32	10.255.255.11	1	100	0	?
*>i	10.255.255.11	1	100	0	?
* i 172.17.255.2/32	10.255.255.22	1	100	0	?
*>i	10.255.255.22	1	100	0	?
*> 172.17.255.3/32	172.17.44.3	1		32768	?

```
SP-PE-R11#show bgp vpnv4 unicast rd 10000:2 172.17.255.2/32
```

```
BGP routing table entry for 10000:2:172.17.255.2/32, version 67
```

```
Paths: (1 available, best #1, table CustB-RIP)
```

```
Advertised to update-groups:
```

```
3
```

```
Refresh Epoch 1
```

```
Local
```

```
10.255.255.22 (metric 20) (via default) from 10.255.255.22 (10.255.255.22)
```

```
Origin incomplete, metric 1, localpref 100, valid, internal, best
```

```
Extended Community: RT:10.255.255.22:2
```

```
mpls labels in/out nolabel/20008
```

```
rx pathid: 0, tx pathid: 0x0
```

```
SP-PE-R11#show bgp vpnv4 unicast rd 10000:2 172.17.255.3/32
```

```
BGP routing table entry for 10000:2:172.17.255.3/32, version 40
```

```
Paths: (1 available, best #1, table CustB-RIP)
```

Advertised to update-groups:

3 4

Refresh Epoch 1

Local, (Received from a RR-client)

10.255.255.44 (metric 20) (via default) from 10.255.255.44 (10.255.255.44)

Origin incomplete, metric 1, localpref 100, valid, internal, best

Extended Community: RT:10.255.255.44:2

mpls labels in/out nolabel/40006

rx pathid: 0, tx pathid: 0x0

CustB-RIP-CE1(config)#do sh ip route | begin Gateway

Gateway of last resort is not set

172.17.0.0/16 is variably subnetted, 7 subnets, 2 masks

C 172.17.11.0/24 is directly connected, Ethernet0/0

L 172.17.11.1/32 is directly connected, Ethernet0/0

R 172.17.22.0/24 [120/1] via 172.17.11.11, 00:00:27, Ethernet0/0

R 172.17.44.0/24 [120/1] via 172.17.11.11, 00:00:27, Ethernet0/0

C 172.17.255.1/32 is directly connected, Loopback0

R 172.17.255.2/32 [120/1] via 172.17.11.11, 00:00:27, Ethernet0/0

R 172.17.255.3/32 [120/1] via 172.17.11.11, 00:00:27, Ethernet0/0

CustB-RIP-CE2(config)#do sh ip route | begin Gate

Gateway of last resort is not set

172.17.0.0/16 is variably subnetted, 7 subnets, 2 masks

R 172.17.11.0/24 [120/1] via 172.17.22.22, 00:00:15, Ethernet0/0

C 172.17.22.0/24 is directly connected, Ethernet0/0

L 172.17.22.2/32 is directly connected, Ethernet0/0

R 172.17.44.0/24 [120/1] via 172.17.22.22, 00:00:15, Ethernet0/0

R 172.17.255.1/32 [120/1] via 172.17.22.22, 00:00:15, Ethernet0/0

C 172.17.255.2/32 is directly connected, Loopback0

R 172.17.255.3/32 [120/1] via 172.17.22.22, 00:00:15, Ethernet0/0

```
CustB-RIP-CE3(config)#do sh ip route | begin Gate
Gateway of last resort is not set

    172.17.0.0/16 is variably subnetted, 7 subnets, 2 masks
R       172.17.11.0/24 [120/1] via 172.17.44.44, 00:00:06, Ethernet1/0
R       172.17.22.0/24 [120/1] via 172.17.44.44, 00:00:06, Ethernet1/0
C       172.17.44.0/24 is directly connected, Ethernet1/0
L       172.17.44.3/32 is directly connected, Ethernet1/0
R       172.17.255.1/32 [120/1] via 172.17.44.44, 00:00:06, Ethernet1/0
R       172.17.255.2/32 [120/1] via 172.17.44.44, 00:00:06, Ethernet1/0
C       172.17.255.3/32 is directly connected, Loopback0
```

```
CustB-RIP-CE1(config)#do trace 172.17.255.2 source lo 0 probe 1
Type escape sequence to abort.
Tracing the route to 172.17.255.2
VRF info: (vrf in name/id, vrf out name/id)
  1 172.17.11.11 2 msec
  2 10.99.11.99 [MPLS: Labels 90002/20008 Exp 0] 2 msec
  3 172.17.22.22 [MPLS: Label 20008 Exp 0] 2 msec
  4 172.17.22.2 2 msec
```

```
CustB-RIP-CE1(config)#do trace 172.17.255.3 source lo 0 probe 1
Type escape sequence to abort.
Tracing the route to 172.17.255.3
VRF info: (vrf in name/id, vrf out name/id)
  1 172.17.11.11 0 msec
  2 10.99.11.99 [MPLS: Labels 90004/40006 Exp 0] 2 msec
  3 172.17.44.44 [MPLS: Label 40006 Exp 0] 1 msec
  4 172.17.44.3 1 msec
```

Let's assume the customer has asked us to block Site 2 and 3 connection but they want connection between site 1 and 2 other sites (Like Hub and spoke connection, Site 1 is the hub):

SP-PE-R22:

```
SP-PE-R22(config-vrf-af)#vrf definition CustB-RIP
SP-PE-R22(config-vrf)#address-family ipv4
SP-PE-R22(config-vrf-af)#no route-target import 10.255.255.44:2
```

SP-PE-R22:

```
SP-PE-R22(config-vrf-af)#vrf definition CustB-RIP
SP-PE-R22(config-vrf)#address-family ipv4
SP-PE-R22(config-vrf-af)#no route-target import 10.255.255.44:2
```

SP-PE-R44:

```
SP-PE-R44(config-vrf-af)#vrf definition CustB-RIP
SP-PE-R44(config-vrf)#address-family ipv4 unicast
SP-PE-R44(config-vrf-af)#no route-target import 10.255.255.22:2
```

**Verification:**

```
CustB-RIP-CE2(config)#do sh ip route rip | begin Gate
Gateway of last resort is not set

    172.17.0.0/16 is variably subnetted, 5 subnets, 2 masks
R       172.17.11.0/24 [120/1] via 172.17.22.22, 00:00:04, Ethernet0/0
R       172.17.255.1/32 [120/1] via 172.17.22.22, 00:00:04, Ethernet0/0

CustB-RIP-CE3(config)#do sh ip route rip | begin Gate
Gateway of last resort is not set

    172.17.0.0/16 is variably subnetted, 5 subnets, 2 masks
R       172.17.11.0/24 [120/1] via 172.17.44.44, 00:00:24, Ethernet1/0
R       172.17.255.1/32 [120/1] via 172.17.44.44, 00:00:24, Ethernet1/0

CustB-RIP-CE1(config)#do sh ip route rip | begin Gate
Gateway of last resort is not set

    172.17.0.0/16 is variably subnetted, 7 subnets, 2 masks
R       172.17.22.0/24 [120/1] via 172.17.11.11, 00:00:06, Ethernet0/0
R       172.17.44.0/24 [120/1] via 172.17.11.11, 00:00:06, Ethernet0/0
R       172.17.255.2/32 [120/1] via 172.17.11.11, 00:00:06, Ethernet0/0
R       172.17.255.3/32 [120/1] via 172.17.11.11, 00:00:06, Ethernet0/0
```

Another usage of having different RTs is Route-Leaking between different VRFs.

If one or more sites of CustA wants to reach an IP address of CustB we can simply use RT import/export to do that:

SP-PE-R22:

```
SP-PE-R11(config)#vrf definition CustA-Static
```

```
SP-PE-R11(config-vrf)#address-family ipv4
```

```
SP-PE-R11(config-vrf-af)#route-target import 10.255.255.22:2
```

```
SP-PE-R11(config-vrf-af)#do sh ip route vrf CustA-Static | begin Gate
```

```
Gateway of last resort is not set
```

```
172.16.0.0/16 is variably subnetted, 7 subnets, 2 masks
```

```
C    172.16.11.0/24 is directly connected, Ethernet0/2
```

```
L    172.16.11.11/32 is directly connected, Ethernet0/2
```

```
B    172.16.22.0/24 [200/0] via 10.255.255.22, 01:56:48
```

```
B    172.16.33.0/24 [200/0] via 10.255.255.33, 01:57:00
```

```
S    172.16.255.1/32 [1/0] via 172.16.11.1, Ethernet0/2
```

```
B    172.16.255.2/32 [200/1] via 10.255.255.22, 01:56:48
```

```
B    172.16.255.3/32 [200/0] via 10.255.255.33, 01:57:00
```

```
172.17.0.0/16 is variably subnetted, 2 subnets, 2 masks
```

```
B    172.17.22.0/24 [200/0] via 10.255.255.22, 00:00:08
```

```
B    172.17.255.2/32 [200/1] via 10.255.255.22, 00:00:08
```



Task 06:

- Provide MPLS L3VPN Service to CustC (Customer C)
- Use BGP as PE-CE Routing Protocol
- Use the same BGP AS#65003 on all Customer Sites. Do not use `allowas-in` option.
- Configure Site-Of-Origin Extended Community for CE2/CE4 site.
- Configure ECMP for VPNv4 traffic of CustC, then CustC-BGP-CE3 traffic must be load-shared between SP-PE-R11 and SP-PE-R33 towards CE4 Lo0.



Solution:

eBGP can also be used as Dynamic PE-CE routing protocol in MPLS L3VPN services. In fact it is the most convenient and easy to configure and manage protocol as a PE-CE routing protocol. There is no need for any redistribution, PE devices will form eBGP peering with CE device and when it gets routes, It will automatically send them to other PE devices using VPNv4 unicast AF.

First Configure Site 1 and Site 3:

CustC-BGP-CE1:

```
interface Loopback0
  ip address 172.18.255.1 255.255.255.255
!
interface Ethernet1/0
  ip address 172.18.11.1 255.255.255.0
!
router bgp 65003
  bgp log-neighbor-changes
  network 172.18.255.1 mask 255.255.255.255
  neighbor 172.18.11.11 remote-as 10000
!
```

CustC-BGP-CE3:

```
interface Loopback0
  ip address 172.18.255.3 255.255.255.255
!
interface Ethernet1/2
  ip address 172.18.44.3 255.255.255.0
```

```
!  
router bgp 65003  
  bgp log-neighbor-changes  
  network 172.18.255.3 mask 255.255.255.255  
  neighbor 172.18.44.44 remote-as 10000  
!
```

CE device configuration is straight forward, the only key point was: the Task asks us to use the same AS# on every Customer site. The problem of using same AS# for all customer site is the Loop prevention mechanism of BGP, CE devices are peering with PE devices using eBGP session, as you already know eBGP peering edge device prepends it's own AS# to the AS_Path attribute then other CE devices receiving these routes just ignore them because they see their own AS# in AS_Path attribute.

To fix this problem we can use **allows-in** option for the neighbor statement, but the Task restricts us to only use other options on Service Provider side.

```
SP-PE-R11:  
vrf definition CustC-BGP  
  rd 10000:65003  
  !  
  address-family ipv4  
    route-target export 10000:3  
    route-target import 10000:3  
  exit-address-family  
!  
!  
interface Ethernet1/0  
  vrf forwarding CustC-BGP  
  ip address 172.18.11.11 255.255.255.0  
!  
router bgp 10000  
!  
  address-family ipv4 vrf CustC-BGP  
    network 172.18.11.0 mask 255.255.255.0  
    neighbor 172.18.11.1 remote-as 65003  
    neighbor 172.18.11.1 activate
```



```
neighbor 172.18.11.1 as-override
exit-address-family
!
SP-PE-R44:
vrf definition CustC-BGP
rd 10000:3
!
address-family ipv4
route-target export 10000:3
route-target import 10000:3
exit-address-family
!
!
interface Ethernet1/2
vrf forwarding CustC-BGP
ip address 172.18.44.44 255.255.255.0
!
router bgp 10000
!
address-family ipv4 vrf CustC-BGP
network 172.18.44.0 mask 255.255.255.0
neighbor 172.18.44.3 remote-as 65003
neighbor 172.18.44.3 activate
neighbor 172.18.44.3 as-override
maximum-paths ibgp 2
exit-address-family
!
```

Look at RD value: **10000:65003** and **10000:3**, I will explain the reason of using different RD values and also **maximum-paths ibgp 2** command at next section of configuration.

PE device must form eBGP peering with CE devices using VRF aware BGP, to do so we configure neighbors and other related commands under address-family ipv4 vrf **CustC-BGP**, there is no need for redistribution because VRF aware BGP gets routes from CE device and sends them to other PE devices using VPNv4 Unicast address-family.

Look at network statements:

```
network 172.18.11.0 mask 255.255.255.0
```

```
network 172.18.44.0 mask 255.255.255.0
```

In fact we don't need these networks (PE-CE connected subnets) to be learned by other Sites, only Loopback interfaces reachability is enough for Site connectivity, but we do it in order to provide Full connectivity between sites. Instead of network statement, redistribute connected command could be used, in that case Origin attribute of NLRI becomes incomplete instead of IGP.

as-override command is used for the neighbors to solve the AS_Path loop prevention problem for our scenario that I've already described. CE devices drop the routes because they see their own AS# in AS_Path. as-override command on Service Provider PE devices replaces Customer AS Number with Provider AS Number when it advertises routes to the customer device, but Provider still sees AS_Path with the actual Customer's AS number.

```
CustC-BGP-CE1#show ip bgp | begin Network
```

	Network	Next Hop	Metric	LocPrf	Weight	Path
r>	172.18.11.0/24	172.18.11.11	0		0	10000 i
*>	172.18.44.0/24	172.18.11.11			0	10000 i
*>	172.18.133.0/24	172.18.11.11			0	10000 i
*>	172.18.233.0/24	172.18.11.11			0	10000 i
*>	172.18.255.1/32	0.0.0.0	0		32768	i
*>	172.18.255.2/32	172.18.11.11			0	10000 10000 i
*>	172.18.255.3/32	172.18.11.11			0	10000 10000 i
*>	172.18.255.4/32	172.18.11.11			0	10000 10000 i

```
SP-PE-R11#show bgp vpnv4 unicast rd 10000:3 | begin Network
```

	Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 10000:3						
*>i	172.18.44.0/24	10.255.255.44	0	100	0	i
*>i	172.18.133.0/24	10.255.255.33	0	100	0	i
*>i	172.18.233.0/24	10.255.255.33	0	100	0	i
*>i	172.18.255.2/32	10.255.255.33	0	100	0	65003 i
*>i	172.18.255.3/32	10.255.255.44	0	100	0	65003 i
*>i	172.18.255.4/32	10.255.255.33	0	100	0	65003 i



Verification:

```
CustC-BGP-CE1#show ip bgp | begin Netw
```

	Network	Next Hop	Metric	LocPrf	Weight	Path
r>	172.18.11.0/24	172.18.11.11	0		0	10000 i
*>	172.18.44.0/24	172.18.11.11			0	10000 i
*>	172.18.255.1/32	0.0.0.0	0		32768	i
*>	172.18.255.3/32	172.18.11.11			0	10000 10000 i

```
CustC-BGP-CE1#trace 172.18.255.3 source lo 0 probe 2
```

Type escape sequence to abort.

Tracing the route to 172.18.255.3

VRF info: (vrf in name/id, vrf out name/id)

```

 1 172.18.11.11 [AS 10000] 1 msec 0 msec
 2 10.99.11.99 [MPLS: Labels 90003/40008 Exp 0] 2 msec 1 msec
 3 172.18.44.44 [AS 10000] [MPLS: Label 40008 Exp 0] 1 msec 1 msec
 4 172.18.44.3 [AS 10000] 1 msec 2 msec
    
```

```
CustC-BGP-CE3#show ip bgp | begin Net
```

	Network	Next Hop	Metric	LocPrf	Weight	Path
*>	172.18.11.0/24	172.18.44.44			0	10000 i
r>	172.18.44.0/24	172.18.44.44	0		0	10000 i
*>	172.18.255.1/32	172.18.44.44			0	10000 10000 i
*>	172.18.255.3/32	0.0.0.0	0		32768	i

```
CustC-BGP-CE3#trace 172.18.255.1 source lo 0 probe 1
```

Type escape sequence to abort.

Tracing the route to 172.18.255.1

VRF info: (vrf in name/id, vrf out name/id)

```

 1 172.18.44.44 [AS 10000] 5 msec
 2 10.99.44.99 [MPLS: Labels 90004/10016 Exp 0] 2 msec
 3 172.18.11.11 [AS 10000] [MPLS: Label 10016 Exp 0] 1 msec
 4 172.18.11.1 [AS 10000] 1 msec
    
```

```
SP-PE-R11(config-if)#$pnv4 uni rd 10000:65003 summary | begin Neighbor
```

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
172.18.11.1	4	65003	7	12	72	0	0	00:03:20	1

```
SP-PE-R44#show bgp vpnv4 uni rd 10000:3 summary | begin Neigh
```

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
172.18.44.3	4	65003	8	12	66	0	0	00:04:06	1

```
SP-PE-R44#show ip cef vrf CustC-BGP 172.18.255.1/32
```

```
172.18.255.1/32
```

```
  nexthop 10.88.44.88 Ethernet1/0 label 80000 10016
```

```
  nexthop 10.99.44.99 Ethernet0/2 label 90001 10016
```

```
SP-PE-R11#show ip cef vrf CustC-BGP 172.18.255.3/32
```

```
172.18.255.3/32
```

```
  nexthop 10.88.11.88 Ethernet0/1 label 80003 40008
```

```
  nexthop 10.99.11.99 Ethernet0/3 label 90004 40008
```

CE2 and CE4 Configuration:

These 2 CE devices are at the same customer site, they are directly connected and also have iBGP peering session with each other:

CustC-BGP-CE2:

```
interface Loopback0
 ip address 172.18.255.2 255.255.255.255
!
interface Ethernet0/0
 ip address 172.18.133.2 255.255.255.0
!
interface Ethernet0/1
 ip address 172.18.24.2 255.255.255.0
!
router bgp 65003
```

```
bgp log-neighbor-changes
network 172.18.255.2 mask 255.255.255.255
neighbor 172.18.133.33 remote-as 10000
neighbor 172.18.255.4 remote-as 65003
neighbor 172.18.255.4 update-source Loopback0
!
ip route 172.18.255.4 255.255.255.255 172.18.24.4
```

CustC-BGP-CE4:

```
interface Loopback0
 ip address 172.18.255.4 255.255.255.255
!
interface Ethernet0/1
 ip address 172.18.24.4 255.255.255.0
!
interface Ethernet0/3
 ip address 172.18.233.4 255.255.255.0
!
interface Ethernet1/2
 ip address 172.18.114.4 255.255.255.0
!
router bgp 65003
 bgp log-neighbor-changes
 network 172.18.255.4 mask 255.255.255.255
 neighbor 172.18.114.11 remote-as 10000
 neighbor 172.18.233.33 remote-as 10000
 neighbor 172.18.255.2 remote-as 65003
 neighbor 172.18.255.2 update-source Loopback0
!
ip route 172.18.255.2 255.255.255.255 172.18.24.2
```

CE2 and CE4 form iBGP peering using their Loopback 0 interfaces, so we need to write a static route for Lo0 networks on each router or we can run an IGP between them. To keep the scenario simple we choose static routes for Lo0 reachability.

Have a close look at the topology, CustC-BGP-CE4 has two physical connections towards SP network, e1/2 (to PE-R11) e0/3 (to PE-R33). Task asks us to load share VPNv4 traffic from site 3 to CE4 device, because of that we have configured different RD value other than 10000:3 on SP-PE-R11, the reason is:

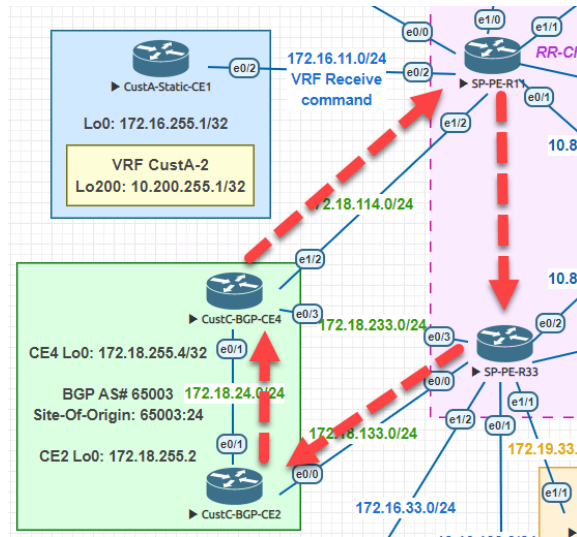
PE devices advertise VPNv4 routes to Route-Reflectors, then Route-Reflectors reflect (send) them to another PE devices, by default a Route-Reflector only advertises the Best route to other PEs, means only a single best route. In this scenario we have two connections from different PE devices to Customer Edge device 4 (CE4), tasks explicitly asks of doing load sharing between two PEs (PE11, PE33), in other words CE3 packets should be load shared between PE11 and PE33. So we need to advertise both routes to PE44, if we set different RDs on different PEs for that customer, then Route-Reflector will send both routes to the PE44:

SP-PE-R11:

```
vrf definition CustC-BGP
  rd 10000:65003
  !
  address-family ipv4
    route-target export 10000:3
    route-target import 10000:3
  exit-address-family
!
!
interface Ethernet1/2
  vrf forwarding CustC-BGP
  ip address 172.18.114.11 255.255.255.0
!
router bgp 10000
!
  address-family ipv4 vrf CustC-BGP
    network 172.18.114.0 mask 255.255.255.0
    neighbor 172.18.114.4 remote-as 65003
    neighbor 172.18.114.4 activate
    neighbor 172.18.114.4 as-override
    neighbor 172.18.114.4 soo 65003:24
  exit-address-family
```

SP-PE-R33:

```
vrf definition CustC-BGP
  rd 10000:3
  !
  address-family ipv4
    route-target export 10000:3
    route-target import 10000:3
  exit-address-family
!
!
interface Ethernet0/0
  vrf forwarding CustC-BGP
  ip address 172.18.133.33 255.255.255.0
!
interface Ethernet0/3
  vrf forwarding CustC-BGP
  ip address 172.18.233.33 255.255.255.0
!
router bgp 10000
  !
  address-family ipv4 vrf CustC-BGP
    network 172.18.133.0 mask 255.255.255.0
    network 172.18.233.0 mask 255.255.255.0
    neighbor 172.18.133.2 remote-as 65003
    neighbor 172.18.133.2 activate
    neighbor 172.18.133.2 as-override
    neighbor 172.18.133.2 soo 65003:24
    neighbor 172.18.233.4 remote-as 65003
    neighbor 172.18.233.4 activate
    neighbor 172.18.233.4 as-override
    neighbor 172.18.233.4 soo 65003:24
  exit-address-family
```



What is SoO here? You noticed that these two CE devices are at the customer site, and they have redundant connections from the Service Provider, every CE device advertises routes to SP PE devices, they can cause a loop if for some reason they advertise back the CE learned routes to another CE device on the same site, and also SP network could become a transit network for CE2-CE4 communication. To avoid these situations every PE must tag routes from the same Customer site with the same value, then PEs simply avoid sending the routes with the same SoO (Site-Of-Origin) extended community value to the CE device.

```
SP-PE-R44#show bgp vpnv4 uni rd 10000:3 | begin Network
```

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 10000:3 (default for vrf CustC-BGP)					
*>i 172.18.11.0/24	10.255.255.11	0	100	0	i
*> 172.18.44.0/24	0.0.0.0	0		32768	i
* i 172.18.133.0/24	10.255.255.33	0	100	0	i
*>i	10.255.255.33	0	100	0	i
* i 172.18.233.0/24	10.255.255.33	0	100	0	i
*>i	10.255.255.33	0	100	0	i
*>i 172.18.255.1/32	10.255.255.11	0	100	0	65003 i
*>i 172.18.255.2/32	10.255.255.11	0	100	0	65003 i
* i	10.255.255.33	0	100	0	65003 i
* i	10.255.255.33	0	100	0	65003 i
*> 172.18.255.3/32	172.18.44.3	0		0	65003 i
*>i 172.18.255.4/32	10.255.255.11	0	100	0	65003 i
* i	10.255.255.33	0	100	0	65003 i
* i	10.255.255.33	0	100	0	65003 i


```
SP-PE-R44#show bgp vpnv4 uni rd 10000:65003 | begin Network
```

Network	Next Hop	Metric	LocPrf	Weight	Path
Route Distinguisher: 10000:65003					
* i 172.18.11.0/24	10.255.255.11	0	100	0	i
*>i	10.255.255.11	0	100	0	i
* i 172.18.255.1/32	10.255.255.11	0	100	0	65003 i
*>i	10.255.255.11	0	100	0	65003 i
* i 172.18.255.2/32	10.255.255.11	0	100	0	65003 i
*>i	10.255.255.11	0	100	0	65003 i
* i 172.18.255.4/32	10.255.255.11	0	100	0	65003 i
*>i	10.255.255.11	0	100	0	65003 i

PE44 receives two routes for network 172.18.255.4/32 with different RDs but puts only one of them to CustC-BGP VRF RIB because of BGP default behavior:

```
SP-PE-R44#show ip route vrf CustC-BGP | begin Gateway
```

```
Gateway of last resort is not set
```

```

172.18.0.0/16 is variably subnetted, 9 subnets, 2 masks
B    172.18.11.0/24 [200/0] via 10.255.255.11, 01:03:31
C    172.18.44.0/24 is directly connected, Ethernet1/2
L    172.18.44.44/32 is directly connected, Ethernet1/2
B    172.18.133.0/24 [200/0] via 10.255.255.33, 01:03:28
B    172.18.233.0/24 [200/0] via 10.255.255.33, 01:03:28
B    172.18.255.1/32 [200/0] via 10.255.255.11, 01:03:31
B    172.18.255.2/32 [200/0] via 10.255.255.11, 00:09:45
B    172.18.255.3/32 [20/0] via 172.18.44.3, 01:03:31
B    172.18.255.4/32 [200/0] via 10.255.255.11, 00:09:45

```

To change BGP's default behavior, an additional command is needed under BGP for Customer VRF:

```
SP-PE-R44:
```

```

router bgp 10000
  address-family ipv4 vrf CustC-BGP
    maximum-paths ibgp 2
  exit-address-family
!
```

Now BGP is able to load share between different paths:

```

SP-PE-R44#show ip route vrf CustC-BGP | begin Gateway
Gateway of last resort is not set

    172.18.0.0/16 is variably subnetted, 9 subnets, 2 masks
B       172.18.11.0/24 [200/0] via 10.255.255.11, 01:08:52
C       172.18.44.0/24 is directly connected, Ethernet1/2
L       172.18.44.44/32 is directly connected, Ethernet1/2
B       172.18.133.0/24 [200/0] via 10.255.255.33, 01:08:49
B       172.18.233.0/24 [200/0] via 10.255.255.33, 01:08:49
B       172.18.255.1/32 [200/0] via 10.255.255.11, 01:08:52
B       172.18.255.2/32 [200/0] via 10.255.255.33, 00:02:45
           [200/0] via 10.255.255.11, 00:02:45
B       172.18.255.3/32 [20/0] via 172.18.44.3, 01:08:52
B       172.18.255.4/32 [200/0] via 10.255.255.33, 00:02:45
           [200/0] via 10.255.255.11, 00:02:45

CustC-BGP-CE3#trace 172.18.255.4 source lo 0 probe 2
Type escape sequence to abort.
Tracing the route to 172.18.255.4
VRF info: (vrf in name/id, vrf out name/id)
  1 172.18.44.44 [AS 10000] 0 msec 1 msec
  2 10.99.44.99 [MPLS: Labels 90003/30010 Exp 0] 2 msec
    10.88.44.88 [MPLS: Labels 80002/30010 Exp 0] 2 msec
  3 172.18.233.33 [AS 10000] [MPLS: Label 30010 Exp 0] 1 msec
    172.18.114.11 [MPLS: Label 10009 Exp 0] 1 msec
  4 172.18.114.4 1 msec
    172.18.233.4 [AS 10000] 1 msec
    
```

To enable Per Micro-Flow load sharing on Cisco IOS enter this command on all routers:

```
ip cef load-sharing algorithm include-ports source destination
```

All Sites have full connectivity:

CustC-BGP-CE4#show ip bgp | Begin Net

	Network	Next Hop	Metric	LocPrf	Weight	Path
* i	172.18.11.0/24	172.18.133.33	0	100	0	10000 i
*		172.18.233.33			0	10000 i
*>		172.18.114.11	0		0	10000 i
*	172.18.44.0/24	172.18.114.11			0	10000 i
* i		172.18.133.33	0	100	0	10000 i
*>		172.18.233.33			0	10000 i
*	172.18.133.0/24	172.18.114.11			0	10000 i
* i		172.18.133.33	0	100	0	10000 i
*>		172.18.233.33	0		0	10000 i
r	172.18.233.0/24	172.18.114.11			0	10000 i
r i		172.18.133.33	0	100	0	10000 i
r>		172.18.233.33	0		0	10000 i
* i	172.18.255.1/32	172.18.133.33	0	100	0	10000 10000 i
*		172.18.233.33			0	10000 10000 i
*>		172.18.114.11			0	10000 10000 i
r>i	172.18.255.2/32	172.18.255.2	0	100	0	i
*	172.18.255.3/32	172.18.114.11			0	10000 10000 i
* i		172.18.133.33	0	100	0	10000 10000 i
*>		172.18.233.33			0	10000 10000 i
*>	172.18.255.4/32	0.0.0.0	0		32768	i

CustC-BGP-CE2#show ip bgp | begin Net

	Network	Next Hop	Metric	LocPrf	Weight	Path
*>	172.18.11.0/24	172.18.133.33			0	10000 i
* i		172.18.114.11	0	100	0	10000 i
* i	172.18.44.0/24	172.18.233.33	0	100	0	10000 i
*>		172.18.133.33			0	10000 i
r i	172.18.133.0/24	172.18.233.33	0	100	0	10000 i
r>		172.18.133.33	0		0	10000 i
* i	172.18.233.0/24	172.18.233.33	0	100	0	10000 i
*>		172.18.133.33	0		0	10000 i
*>	172.18.255.1/32	172.18.133.33			0	10000 10000 i
* i		172.18.114.11	0	100	0	10000 10000 i
*>	172.18.255.2/32	0.0.0.0	0		32768	i
* i	172.18.255.3/32	172.18.233.33	0	100	0	10000 10000 i
*>		172.18.133.33			0	10000 10000 i
r>i	172.18.255.4/32	172.18.255.4	0	100	0	i



Task 07:

- Provide MPLS L3VPN Service to CustD (Customer D)
- Use OSPF as PE-CE Routing Protocol
- Configure domain-id type 0005 value 000000065004 for OSPF process
- Sites should use optimal paths in order to reach each other (Serial WAN links should be used as a backup path as long as PE-CE connections are up)
- CE5 int e0/0, Lo0 and also OSPF process should be configured under a VRF



Solution:

First configure CE devices, they have a straight forward configuration other than CE5, so we only configure CE1-4 for now:

CustD-OSPF-CE1:

```
interface Loopback0
 ip address 172.19.255.1 255.255.255.255
 ip ospf 4 area 0
!
interface Ethernet1/1
 ip address 172.19.33.1 255.255.255.0
 ip ospf network point-to-point
 ip ospf 4 area 0
!
interface Serial2/0
 ip address 172.19.12.1 255.255.255.0
 ip ospf network point-to-point
 ip ospf 4 area 0
 serial restart-delay 0
!
router ospf 4
 passive-interface default
 no passive-interface Ethernet1/1
 no passive-interface Serial2/0
!
```

CustD-OSPF-CE2:

```
interface Loopback0
  ip address 172.19.255.2 255.255.255.255
  ip ospf 4 area 0
!
interface Ethernet0/3
  ip address 172.19.244.2 255.255.255.0
  ip ospf network point-to-point
  ip ospf 4 area 0
!
interface Serial2/0
  ip address 172.19.12.2 255.255.255.0
  ip ospf network point-to-point
  ip ospf 4 area 0
  serial restart-delay 0
!
router ospf 4
  passive-interface default
  no passive-interface Ethernet0/3
  no passive-interface Serial2/0
!
```

CustD-OSPF-CE3:

```
interface Loopback0
  ip address 172.19.255.3 255.255.255.255
  ip ospf 4 area 0
!
interface Ethernet1/1
  ip address 172.19.11.3 255.255.255.0
  ip ospf network point-to-point
  ip ospf 4 area 0
!
interface Serial2/0
```

```
ip address 172.19.34.3 255.255.255.0
ip ospf network point-to-point
ip ospf 4 area 0
serial restart-delay 0
!
router ospf 4
  passive-interface default
  no passive-interface Ethernet1/1
  no passive-interface Serial2/0
!
```

CustD-OSPF-CE4:

```
interface Loopback0
  ip address 172.19.255.4 255.255.255.255
  ip ospf 4 area 0
!
interface Ethernet1/1
  ip address 172.19.22.4 255.255.255.0
  ip ospf network point-to-point
  ip ospf 4 area 0
!
interface Serial2/0
  ip address 172.19.34.4 255.255.255.0
  ip ospf network point-to-point
  ip ospf 4 area 0
  serial restart-delay 0
!
router ospf 4
  passive-interface default
  no passive-interface Ethernet1/1
  no passive-interface Serial2/0
!
```

Customer site routers are OSPF neighbors now:

```
CustD-OSPF-CE1#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
172.19.255.2	0	FULL/ -	00:00:30	172.19.12.2	Serial2/0

```
CustD-OSPF-CE2#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
172.19.255.1	0	FULL/ -	00:00:39	172.19.12.1	Serial2/0

```
CustD-OSPF-CE3#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
172.19.255.4	0	FULL/ -	00:00:32	172.19.34.4	Serial2/0

```
CustD-OSPF-CE4#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
172.19.255.3	0	FULL/ -	00:00:35	172.19.34.3	Serial2/0

Configure PE side:

SP-PE-R11:

```
vrf definition CustD-OSPF
 rd 10000:4
 !
 address-family ipv4
  route-target export 10000:4
  route-target import 10000:4
 exit-address-family
 !
 !
 interface Ethernet1/1
  vrf forwarding CustD-OSPF
```

```
ip address 172.19.11.11 255.255.255.0
ip ospf network point-to-point
!
interface Loopback65004
 vrf forwarding CustD-OSPF
 ip address 172.19.255.11 255.255.255.255
!
router ospf 65004 vrf CustD-OSPF
 router-id 172.19.255.11
 domain-id type 0005 value 000000065004
 redistribute bgp 10000 subnets
 network 172.19.11.11 0.0.0.0 area 0
!
!
router bgp 10000
!
 address-family ipv4 vrf CustD-OSPF
  redistribute ospf 65004 match internal external 1 external 2
 exit-address-family
!

SP-PE-R22:
vrf definition CustD-OSPF
 rd 10000:4
!
 address-family ipv4
  route-target export 10000:4
  route-target import 10000:4
 exit-address-family
!
!
interface Ethernet1/1
 vrf forwarding CustD-OSPF
 ip address 172.19.22.22 255.255.255.0
 ip ospf network point-to-point
```



```

!
interface Loopback65004
 vrf forwarding CustD-OSPF
 ip address 172.19.255.22 255.255.255.255
!
router ospf 65004 vrf CustD-OSPF
 router-id 172.19.255.22
 domain-id type 0005 value 000000065004
 redistribute bgp 10000 subnets
 network 172.19.22.22 0.0.0.0 area 0
!
!
router bgp 10000
!
 address-family ipv4 vrf CustD-OSPF
  redistribute ospf 65004 match internal external 1 external 2
 exit-address-family
!
    
```

Most configurations are the same as previous tasks but there are some additional command that we have highlighted:

IOS/IOS-XE sets domain-id value to based on OSPF Process ID, this value is used for some reason that will be explained in next pages.

IOS-XR does not set domain-id automatically, you must set it explicitly. There is no IOS-XR device in this lab but you will see XRv routers in next labs, and should be aware of this default behavior.

```

CustD-OSPF-CE3#show ip route ospf | begin Gate
Gateway of last resort is not set
    172.19.0.0/16 is variably subnetted, 11 subnets, 2 masks
O        172.19.22.0/24 [110/74] via 172.19.34.4, 00:00:17, Serial2/0
O        172.19.255.4/32 [110/65] via 172.19.34.4, 00:00:17, Serial2/0

CustD-OSPF-CE4#show ip route ospf | begin Gate
Gateway of last resort is not set
    
```

```

172.19.0.0/16 is variably subnetted, 11 subnets, 2 masks
0       172.19.11.0/24 [110/74] via 172.19.34.3, 00:00:34, Serial2/0
0       172.19.255.3/32 [110/65] via 172.19.34.3, 00:00:34, Serial2/0
    
```

Why CE3 and CE4 use Serial2/0 WAN connection to reach each other Loopback 0?

Let's shutdown Serial 2/0 interface to see what happens:

```

CustD-OSPF-CE4(config)#int s2/0
CustD-OSPF-CE4(config-if)#shu
CustD-OSPF-CE4(config-if)#shutdown
CustD-OSPF-CE4(config-if)#
*Apr  6 15:21:00.142: %OSPF-5-ADJCHG: Process 4, Nbr 172.19.255.3 on Serial2/0 from FULL to DOWN, Neighbor
Down: Interface down or detached
CustD-OSPF-CE4(config-if)#do
*Apr  6 15:21:02.142: %LINK-5-CHANGED: Interface Serial2/0, changed state to administratively down
*Apr  6 15:21:03.146: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to down
CustD-OSPF-CE4(config-if)#do sh ip route ospf | begin Gate
Gateway of last resort is not set

172.19.0.0/16 is variably subnetted, 10 subnets, 2 masks
0 IA    172.19.11.0/24 [110/11] via 172.19.22.22, 00:00:11, Ethernet1/1
0 IA    172.19.34.0/24 [110/84] via 172.19.22.22, 00:00:11, Ethernet1/1
0 IA    172.19.255.3/32 [110/21] via 172.19.22.22, 00:00:11, Ethernet1/1
    
```

Interesting! SP advertised routes are Inter-Area routes, but all interfaces of all PE and CE routers are in Area 0, What are those Inter-Area routes? We don't have any other Areas than area 0.

This is because of domain-id configuration on PE routers, If OSPF DOMAIN ID extended community value of a route is the same as the value we configured on every PE device, routes become Inter-Area:

```

SP-PE-R11#show bgp vpnv4 unicast rd 10000:4 172.19.255.4/32
BGP routing table entry for 10000:4:172.19.255.4/32, version 96
Paths: (1 available, best #1, table CustD-OSPF)
  Advertised to update-groups:
    3
  Refresh Epoch 1
  Local
    
```

```
10.255.255.22 (metric 20) (via default) from 10.255.255.22 (10.255.255.22)
  Origin incomplete, metric 11, localpref 100, valid, internal, best
  Extended Community: RT:10000:4 OSPF DOMAIN ID:0x0005:0x000000065004
    OSPF RT:0.0.0.0:2:0 OSPF ROUTER ID:172.19.255.22:0
  mpls labels in/out nolabel/20013
  rx pathid: 0, tx pathid: 0x0
```

Let's change Domain-ID value on PE22 and check the result again:

```
SP-PE-R22(config)#router ospf 65004 vrf CustD-OSPF
SP-PE-R22(config-router)#domain-id type 0005 value 000000065000

CustD-OSPF-CE4(config-if)#do sh ip route ospf | begin Gate
Gateway of last resort is not set

    172.19.0.0/16 is variably subnetted, 9 subnets, 2 masks
O E2    172.19.11.0/24 [110/1] via 172.19.22.22, 00:00:42, Ethernet1/1
O E2    172.19.255.3/32 [110/11] via 172.19.22.22, 00:00:42, Ethernet1/1
```

Routes became External Type-2.

Time to figure out what Sham-Link is:

The task asks us to configure sites in a way that they use only optimal paths, but using 1.54 Mb/s Serial WAN link is not optimal because PE-CE links are ethernet so they have a bandwidth of 10 Mb/s.

The reason that Serial link is preferred over an Ethernet link is because:

OSPF prefers Intra-Area routes to Inter-Area routes:

```
CustD-OSPF-CE3(config)#do show ip route ospf | begin Gate
Gateway of last resort is not set

    172.19.0.0/16 is variably subnetted, 9 subnets, 2 masks
O IA    172.19.22.0/24 [110/11] via 172.19.11.11, 00:03:50, Ethernet1/1
O IA    172.19.255.4/32 [110/21] via 172.19.11.11, 00:03:50, Ethernet1/1
CustD-OSPF-CE3(config)#int s2/0
CustD-OSPF-CE3(config-if)#no sh
*Apr  6 15:37:10.435: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
*Apr  6 15:37:10.473: %OSPF-5-ADJCHG: Process 4, Nbr 172.19.255.4 on Serial2/0 from LOADING to FULL, Loading Done
```

```
CustD-OSPF-CE3(config-if)#do show ip route ospf | begin Gate
Gateway of last resort is not set

    172.19.0.0/16 is variably subnetted, 11 subnets, 2 masks
0       172.19.22.0/24 [110/74] via 172.19.34.4, 00:00:04, Serial2/0
0       172.19.255.4/32 [110/65] via 172.19.34.4, 00:00:04, Serial2/0
```

We have to convert SP learned routes to Intra-Area routes. But how? Can we make SP network between PE11 and PE22 Backbone? Using Sham-Link yes! SP network becomes Backbone for our Backbones! This is called Super-Backbone.

Create a loopback interface on each PE router and advertise them into the BGP:

SP-PE-R11:

```
interface Loopback65004
 vrf forwarding CustD-OSPF
 ip address 172.19.255.11 255.255.255.255
!
router ospf 65004 vrf CustD-OSPF
 area 0 sham-link 172.19.255.11 172.19.255.22
 network 172.19.11.11 0.0.0.0 area 0
!
router bgp 10000
!
 address-family ipv4 vrf CustD-OSPF
  network 172.19.255.11 mask 255.255.255.255
```

SP-PE-R22:

```
interface Loopback65004
 vrf forwarding CustD-OSPF
 ip address 172.19.255.22 255.255.255.255
!
router ospf 65004 vrf CustD-OSPF
 area 0 sham-link 172.19.255.22 172.19.255.11
```

```

!
router bgp 10000
!
address-family ipv4 vrf CustD-OSPF
  network 172.19.255.22 mask 255.255.255.255
!
    
```

Now PE11 and PE22 can ping each other Loopback 65004:

```

SP-PE-R11#show bgp vpnv4 unicast rd 10000:4 | include 172.19.255.22
 *>i 172.19.255.22/32 10.255.255.22          0    100    0 i

SP-PE-R22#show bgp vpnv4 unicast rd 10000:4 | include 172.19.255.11
 *>i 172.19.255.11/32 10.255.255.11       0    100    0 i

SP-PE-R11#ping vrf CustD-OSPF 172.19.255.22 source lo 65004
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.19.255.22, timeout is 2 seconds:
Packet sent with a source address of 172.19.255.11
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
    
```

They can form Sham-Link using their Loopback 65004 interfaces:

```

SP-PE-R11(config)#router ospf 65004 vrf CustD-OSPF
SP-PE-R11(config-router)#area 0 sham-link 172.19.255.11 172.19.255.22
SP-PE-R22(config)#router ospf 65004 vrf CustD-OSPF
SP-PE-R22(config-router)#area 0 sham-link 172.19.255.22 172.19.255.11
    
```

```

SP-PE-R22#show ip ospf 65004 neighbor
Neighbor ID    Pri  State           Dead Time   Address        Interface
172.19.255.11  0    FULL/ -         -           172.19.255.11  OSPF_SL1
172.19.255.4   0    FULL/ -         00:00:33    172.19.22.4   Ethernet1/1
SP-PE-R11#show ip ospf 65004 neighbor
Neighbor ID    Pri  State           Dead Time   Address        Interface
172.19.255.22  0    FULL/ -         -           172.19.255.22  OSPF_SL1
172.19.255.3   0    FULL/ -         00:00:37    172.19.11.3   Ethernet1/1
    
```

This Area 0 using Sham-Link on PE Routers acts like a Super-Backbone area for OSPF process on Customer sites, and Ethernet1/1 is preferred over Serial 2/0:

```
CustD-OSPF-CE3#show ip route ospf | begin Gate
Gateway of last resort is not set

    172.19.0.0/16 is variably subnetted, 11 subnets, 2 masks
O       172.19.22.0/24 [110/21] via 172.19.11.11, 00:03:20, Ethernet1/1
O       172.19.255.4/32 [110/22] via 172.19.11.11, 00:03:20, Ethernet1/1
O E2    172.19.255.11/32 [110/1] via 172.19.11.11, 01:20:43, Ethernet1/1
O E2    172.19.255.22/32 [110/1] via 172.19.11.11, 01:19:36, Ethernet1/1
```

```
CustD-OSPF-CE4#show ip route ospf | begin Gate
Gateway of last resort is not set

    172.19.0.0/16 is variably subnetted, 11 subnets, 2 masks
O       172.19.11.0/24 [110/21] via 172.19.22.22, 00:04:17, Ethernet1/1
O       172.19.255.3/32 [110/22] via 172.19.22.22, 00:04:17, Ethernet1/1
O E2    172.19.255.11/32 [110/1] via 172.19.22.22, 01:20:33, Ethernet1/1
O E2    172.19.255.22/32 [110/1] via 172.19.22.22, 01:21:49, Ethernet1/1
```

Lets configure other 2 PE devices the same way:

```
SP-PE-R33:
vrf definition CustD-OSPF
rd 10000:4
!
address-family ipv4
route-target export 10000:4
route-target import 10000:4
exit-address-family
!
!
interface Ethernet1/1
vrf forwarding CustD-OSPF
ip address 172.19.33.33 255.255.255.0
```

```
ip ospf network point-to-point
shutdown
!
interface Loopback65004
 vrf forwarding CustD-OSPF
 ip address 172.19.255.33 255.255.255.255
!
router ospf 65004 vrf CustD-OSPF
 router-id 172.19.255.33
 domain-id type 0005 value 000000065004
 area 0 sham-link 172.19.255.33 172.19.255.44
 redistribute bgp 10000 subnets
 network 172.19.33.33 0.0.0.0 area 0
!
!
router bgp 10000
!
 address-family ipv4 vrf CustD-OSPF
  network 172.19.255.33 mask 255.255.255.255
  redistribute ospf 65004 match internal external 1 external 2
 exit-address-family
!
```

SP-PE-R44:

```
vrf definition CustD-OSPF
 rd 10000:4
!
 address-family ipv4
  route-target export 10000:4
  route-target import 10000:4
 exit-address-family
!
!
interface Ethernet0/0
```

```
vrf forwarding CustD-OSPF
ip address 172.19.144.44 255.255.255.0
ip ospf network point-to-point
shutdown
!
interface Ethernet0/3
vrf forwarding CustD-OSPF
ip address 172.19.244.44 255.255.255.0
ip ospf network point-to-point
shutdown
!
interface Loopback65004
vrf forwarding CustD-OSPF
ip address 172.19.255.44 255.255.255.255
!
router ospf 65004 vrf CustD-OSPF
router-id 172.19.255.44
domain-id type 0005 value 000000065004
area 0 sham-link 172.19.255.44 172.19.255.33
redistribute bgp 10000 subnets
network 172.19.144.44 0.0.0.0 area 0
network 172.19.244.44 0.0.0.0 area 0
!
router bgp 10000
!
address-family ipv4 vrf CustD-OSPF
network 172.19.255.44 mask 255.255.255.255
redistribute ospf 65004 match internal external 1 external 2
exit-address-family
!
```

One more step to complete the task!

CE5 Configuration, the Task asks us to configure CE5 using a VRF not global routing table:

CustD-OSPF-CE5:

```

vrf definition CustD
!
 address-family ipv4
  exit-address-family
!
interface Loopback0
 vrf forwarding CustD
 ip address 172.19.255.5 255.255.255.255
 ip ospf 4 area 0
!
interface Ethernet0/0
 vrf forwarding CustD
 ip address 172.19.144.5 255.255.255.0
 ip ospf network point-to-point
 ip ospf 4 area 0
!
router ospf 4 vrf CustD
 router-id 172.19.255.5
!
    
```

Lets verify CE5 connectivity:

```

CustD-OSPF-CE5#show ip route vrf CustD
Routing Table: CustD
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override
Gateway of last resort is not set

172.19.0.0/16 is variably subnetted, 10 subnets, 2 masks
    
```

```

O    172.19.12.0/24 [110/84] via 172.19.144.44, 00:01:56, Ethernet0/0
O    172.19.33.0/24 [110/21] via 172.19.144.44, 00:01:56, Ethernet0/0
C    172.19.144.0/24 is directly connected, Ethernet0/0
L    172.19.144.5/32 is directly connected, Ethernet0/0
O    172.19.244.0/24 [110/20] via 172.19.144.44, 00:01:56, Ethernet0/0
O    172.19.255.1/32 [110/22] via 172.19.144.44, 00:01:56, Ethernet0/0
O    172.19.255.2/32 [110/21] via 172.19.144.44, 00:01:56, Ethernet0/0
C    172.19.255.5/32 is directly connected, Loopback0
O E2  172.19.255.33/32 [110/1] via 172.19.144.44, 00:01:56, Ethernet0/0
O E2  172.19.255.44/32 [110/1] via 172.19.144.44, 00:01:56, Ethernet0/0
    
```

RIB is incomplete, some routes are missing! But what about OSPF LS Database? It is complete:

```
CustD-OSPF-CE5#show ip ospf 4 database
```

```
    OSPF Router with ID (172.19.255.5) (Process ID 4)
```

```
    Router Link States (Area 0)
```

Link ID	ADV Router	Age	Seq#	Checksum	Link count
172.19.255.1	172.19.255.1	325	0x8000002C	0x00F6FE	5
172.19.255.2	172.19.255.2	614	0x8000002D	0x00D667	5
172.19.255.5	172.19.255.5	185	0x80000005	0x006AB0	3
172.19.255.33	172.19.255.33	1160	0x80000030	0x00075E	3
172.19.255.44	172.19.255.44	372	0x80000035	0x002BB2	5

```
    Summary Net Link States (Area 0)
```

Link ID	ADV Router	Age	Seq#	Checksum
172.19.11.0	172.19.255.33	147	0x80000029	0x0027C0
172.19.11.0	172.19.255.44	616	0x80000029	0x00E4F7
172.19.22.0	172.19.255.33	1416	0x80000027	0x00B12D
172.19.22.0	172.19.255.44	1374	0x80000027	0x006F64
172.19.34.0	172.19.255.33	1160	0x80000027	0x000A7F
172.19.34.0	172.19.255.44	1117	0x80000027	0x00C7B6
172.19.255.3	172.19.255.33	147	0x80000029	0x00E6FE
172.19.255.3	172.19.255.44	616	0x80000029	0x00A436
172.19.255.4	172.19.255.33	1416	0x80000027	0x00E006

```
172.19.255.4    172.19.255.44    1374          0x80000027 0x009E3D
```

Type-5 AS External Link States

Link ID	ADV Router	Age	Seq#	Checksum	Tag
172.19.255.11	172.19.255.33	147	0x80000029	0x00BE98	3489670928
172.19.255.11	172.19.255.44	616	0x80000029	0x007CCF	3489670928
172.19.255.22	172.19.255.33	147	0x80000029	0x0050FB	3489670928
172.19.255.22	172.19.255.44	616	0x80000029	0x000E33	3489670928
172.19.255.33	172.19.255.33	147	0x80000029	0x006958	3489670928
172.19.255.33	172.19.255.44	616	0x80000029	0x009F96	3489670928
172.19.255.44	172.19.255.33	147	0x80000029	0x0073C2	3489670928
172.19.255.44	172.19.255.44	616	0x80000029	0x00B8F2	3489670928

Link-State Database is fully collected, but some routes are not imported to the RIB. The reason is DN-bit or Downward bit. It is a loop prevention mechanism of OSPF in MPLS VPNs. When a PE router imports OSPF learned routes from VPNv4 AF (Inter-Area and External routes, not Sham-Link learned routes), it sets Downward bit for those LSAs, then other VRF aware OSPF processes on another routers ignore those LSAs and never import the routes to the RIB. It is a loop prevention mechanism of VRF aware OSPF. When we configure normal OSPF process on a CE device, these downward bit tagged LSAs are accepted from PE device, but if we run VRF aware OSPF Process on CE devices, downward bit tagged LSAs are ignored by CE device and never get into RIB.

Some routes are imported to CE device's RIB, because they don't have Downward bit.

To fix this problem we need to enter a command on CE device under the OSPF process, then CE device ignores DN-bit:

```
CustD-OSPF-CE5:
router ospf 4 vrf CustD
  capability vrf-lite
!
```

Now RIB is complete:

```
CustD-OSPF-CE5#show ip ospf database
```

```
OSPF Router with ID (172.19.255.5) (Process ID 4)
```

```
Router Link States (Area 0)
```

Link ID	ADV Router	Age	Seq#	Checksum	Link count
172.19.255.1	172.19.255.1	1238	0x8000002C	0x00F6FE	5
172.19.255.2	172.19.255.2	1527	0x8000002D	0x00D667	5
172.19.255.5	172.19.255.5	247	0x80000006	0x0065B5	3
172.19.255.33	172.19.255.33	68	0x80000031	0x00055F	3
172.19.255.44	172.19.255.44	1285	0x80000035	0x002BB2	5

```
Summary Net Link States (Area 0)
```

Link ID	ADV Router	Age	Seq#	Checksum
172.19.11.0	172.19.255.33	1059	0x80000029	0x0027C0
172.19.11.0	172.19.255.44	1529	0x80000029	0x00E4F7
172.19.22.0	172.19.255.33	314	0x80000028	0x00AF2E
172.19.22.0	172.19.255.44	292	0x80000028	0x006D65
172.19.34.0	172.19.255.33	68	0x80000028	0x000880
172.19.34.0	172.19.255.44	39	0x80000028	0x00C5B7
172.19.255.3	172.19.255.33	1059	0x80000029	0x00E6FE
172.19.255.3	172.19.255.44	1529	0x80000029	0x00A436
172.19.255.4	172.19.255.33	314	0x80000028	0x00DE07
172.19.255.4	172.19.255.44	292	0x80000028	0x009C3E

```
Type-5 AS External Link States
```

Link ID	ADV Router	Age	Seq#	Checksum	Tag
172.19.255.11	172.19.255.33	1059	0x80000029	0x00BE98	3489670928
172.19.255.11	172.19.255.44	1529	0x80000029	0x007CCF	3489670928
172.19.255.22	172.19.255.33	1059	0x80000029	0x0050FB	3489670928
172.19.255.22	172.19.255.44	1529	0x80000029	0x000E33	3489670928
172.19.255.33	172.19.255.33	1059	0x80000029	0x006958	3489670928

```

172.19.255.33 172.19.255.44 1529 0x80000029 0x009F96 3489670928
172.19.255.44 172.19.255.33 1059 0x80000029 0x0073C2 3489670928
172.19.255.44 172.19.255.44 1529 0x80000029 0x00B8F2 3489670928
    
```

```
CustD-OSPF-CE5#show ip route vrf CustD | begin Gate
```

```
Gateway of last resort is not set
```

```
172.19.0.0/16 is variably subnetted, 17 subnets, 2 masks
```

```

O IA 172.19.11.0/24 [110/11] via 172.19.144.44, 00:05:14, Ethernet0/0
O    172.19.12.0/24 [110/84] via 172.19.144.44, 00:05:14, Ethernet0/0
O IA 172.19.22.0/24 [110/11] via 172.19.144.44, 00:05:14, Ethernet0/0
O    172.19.33.0/24 [110/21] via 172.19.144.44, 00:05:14, Ethernet0/0
O IA 172.19.34.0/24 [110/84] via 172.19.144.44, 00:05:14, Ethernet0/0
C    172.19.144.0/24 is directly connected, Ethernet0/0
L    172.19.144.5/32 is directly connected, Ethernet0/0
O    172.19.244.0/24 [110/20] via 172.19.144.44, 00:05:14, Ethernet0/0
O    172.19.255.1/32 [110/22] via 172.19.144.44, 00:05:14, Ethernet0/0
O    172.19.255.2/32 [110/21] via 172.19.144.44, 00:05:14, Ethernet0/0
O IA 172.19.255.3/32 [110/21] via 172.19.144.44, 00:05:14, Ethernet0/0
O IA 172.19.255.4/32 [110/21] via 172.19.144.44, 00:05:14, Ethernet0/0
C    172.19.255.5/32 is directly connected, Loopback0
O E2 172.19.255.11/32 [110/1] via 172.19.144.44, 00:05:14, Ethernet0/0
O E2 172.19.255.22/32 [110/1] via 172.19.144.44, 00:05:14, Ethernet0/0
O E2 172.19.255.33/32 [110/1] via 172.19.144.44, 00:05:14, Ethernet0/0
O E2 172.19.255.44/32 [110/1] via 172.19.144.44, 00:05:14, Ethernet0/0
    
```

```
CustD-OSPF-CE5#trace vrf CustD 172.19.255.3 source lo 0
```

```
Type escape sequence to abort.
```

```
Tracing the route to 172.19.255.3
```

```
VRF info: (vrf in name/id, vrf out name/id)
```

```

 1 172.19.144.44 1 msec 1 msec 1 msec
 2 10.88.44.88 [MPLS: Labels 80000/10011 Exp 0] 1 msec 2 msec 1 msec
 3 172.19.11.11 [MPLS: Label 10011 Exp 0] 1 msec 2 msec 1 msec
 4 172.19.11.3 1 msec 1 msec 1 msec
    
```



Task 07:

- Customer B wants to have IPv6 connectivity, Provide it using 6VPE service
- Configure OSPFv3 as the PE-CE routing protocol
- Physical Interface IPv6 addressing scheme must be:

FD02:172:17:PE_R#::R#/64

- Loopback Interface IPv6 addressing scheme must be:

FD02:172:17:255::R#/128



Solution:

First, configure CE devices:

CustB-RIP-CE1:

```
ipv6 unicast-routing
!
interface Loopback0
 ip address 172.17.255.1 255.255.255.255
 ipv6 address FD02:172:17:255::1/128
 ospfv3 2 ipv6 area 0
!
interface Ethernet0/0
 ip address 172.17.11.1 255.255.255.0
 ipv6 address FD02:172:17:11::1/64
 ospfv3 2 ipv6 area 0
!
```

CustB-RIP-CE2:

```
ipv6 unicast-routing
!
interface Loopback0
 ip address 172.17.255.1 255.255.255.255
 ipv6 address FD02:172:17:255::2/128
 ospfv3 2 ipv6 area 0
!
```

```
interface Ethernet0/0
 ip address 172.17.11.1 255.255.255.0
 ipv6 address FD02:172:17:22::2/64
 ospfv3 2 ipv6 area 0
!

CustB-RIP-CE3:
ipv6 unicast-routing
!
interface Loopback0
 ip address 172.17.255.1 255.255.255.255
 ipv6 address FD02:172:17:255::3/128
 ospfv3 2 ipv6 area 0
!
interface Ethernet1/0
 ip address 172.17.11.1 255.255.255.0
 ipv6 address FD02:172:17:44::3/64
 ospfv3 2 ipv6 area 0
!
```

IPv6 Unicast routing and also OSPFv3 must be enable on CE devices because RIP does not support IPv4 Address-Family and also VRF aware RIPng is not supported in IOS.

Between PE devices, iBGP session using IPv4 Unicast addresses is enough for 6VPE service, the Underlay connectivity is going to be IPv4 only.

We need to also enable VPNv6 Address-Family between PE devices:

```
SP-PE-R11:
ipv6 unicast-routing
router bgp 10000
 address-family vpnv6
  neighbor 10.255.255.22 activate
  neighbor 10.255.255.22 send-community extended
  neighbor 10.255.255.44 activate
  neighbor 10.255.255.44 send-community extended
  neighbor 10.255.255.44 route-reflector-client
!
```

SP-PE-R22:

```
ipv6 unicast-routing
router bgp 10000
address-family vpnv6
neighbor 10.255.255.11 activate
neighbor 10.255.255.11 send-community extended
neighbor 10.255.255.44 activate
neighbor 10.255.255.44 send-community extended
neighbor 10.255.255.44 route-reflector-client
!
```

SP-PE-R44:

```
ipv6 unicast-routing
router bgp 10000
address-family vpnv6
neighbor 10.255.255.11 activate
neighbor 10.255.255.11 send-community extended
neighbor 10.255.255.22 activate
neighbor 10.255.255.22 send-community extended
!
```

**Verification:**

```
SP-PE-R11(config-router)#do sh bgp vpnv6 uni all summ | begin Ne
```

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.255.255.22	4	10000	42	44	1	0	0	00:02:09	0
10.255.255.44	4	10000	13	43	1	0	0	00:01:09	0

```
SP-PE-R22(config-router-af)#do sh bgp vpnv6 uni all summ | begin Ne
```

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.255.255.11	4	10000	44	42	1	0	0	00:02:19	0
10.255.255.44	4	10000	15	42	1	0	0	00:01:19	0


```

SP-PE-R44(config-router-af)#do sh bgp vpnv6 uni all summ | begin Ne
Neighbor          V          AS MsgRcvd MsgSent   TblVer   InQ  OutQ Up/Down   State/PfxRcd
10.255.255.11     4          10000    43     13         1     0    0 00:01:38      0
10.255.255.22     4          10000    42     15         1     0    0 00:01:38      0
    
```

MP-BGP VPNv6 Unicast peering has been established, time to configure PE devices to communicate with CE devices and transmit their IPv6 routes:

SP-PE-R11:

```

vrf definition CustB-RIP
!
address-family ipv6
    route-target export 10000:2
    route-target import 10000:2
exit-address-family
!
!
interface Ethernet0/0
    vrf forwarding CustB-RIP
    ipv6 address FD02:172:17:11::11/64
    ospfv3 2 ipv6 area 0
!
router ospfv3 2
!
address-family ipv6 unicast
exit-address-family
!
address-family ipv6 unicast vrf CustB-RIP
    redistribute bgp 10000
exit-address-family
!
router bgp 10000
!
address-family ipv6 vrf CustB-RIP
    redistribute ospf 2 match internal external 1 external 2
exit-address-family
!
    
```

SP-PE-R22:

```
vrf definition CustB-RIP
!
  address-family ipv6
    route-target export 10000:2
    route-target import 10000:2
  exit-address-family
!
!
interface Ethernet0/0
  vrf forwarding CustB-RIP
  ipv6 address FD02:172:17:22::22/64
  ospfv3 2 ipv6 area 0
!
router ospfv3 2
!
  address-family ipv6 unicast
  exit-address-family
!
  address-family ipv6 unicast vrf CustB-RIP
    redistribute bgp 10000
  exit-address-family
!
router bgp 10000
!
  address-family ipv6 vrf CustB-RIP
    redistribute ospf 2 match internal external 1 external 2
  exit-address-family
!
```

SP-PE-R44:

```
vrf definition CustB-RIP
!
  address-family ipv6
    route-target export 10000:2
```

```

route-target import 10000:2
exit-address-family
!
interface Ethernet0/1
vrf forwarding CustB-RIP
ipv6 address FD02:172:17:44::44/64
ospfv3 2 ipv6 area 0
!
router ospfv3 2
!
address-family ipv6 unicast
exit-address-family
!
address-family ipv6 unicast vrf CustB-RIP
redistribute bgp 10000
exit-address-family
!
router bgp 10000
!
address-family ipv6 vrf CustB-RIP
redistribute ospf 2 match internal external 1 external 2
exit-address-family
!
    
```

Verification:

```

SP-PE-R11(config)#do sh bgp vpnv6 uni rd 10000:2 | begin Net
    Network          Next Hop          Metric LocPrf Weight Path
Route Distinguisher: 10000:2 (default for vrf CustB-RIP)
*>  FD02:172:17:255::1/128
           FE80::A8BB:CCFF:FE00:700
                                   10          32768 ?
*>i  FD02:172:17:255::2/128
           ::FFFF:10.255.255.22
                                   10    100    0 ?
*>i  FD02:172:17:255::3/128
    
```

```
                ::FFFF:10.255.255.44
                                10    100    0 ?
SP-PE-R11(config)#do sh bgp vpnv6 uni rd 10000:2  FD02:172:17:255::2/128
BGP routing table entry for [10000:2]FD02:172:17:255::2/128, version 46
Paths: (1 available, best #1, table CustB-RIP)
  Advertised to update-groups:
    6
  Refresh Epoch 5
  Local
    ::FFFF:10.255.255.22 (metric 20) (via default) from 10.255.255.22 (10.255.255.22)
    Origin incomplete, metric 10, localpref 100, valid, internal, best
    Extended Community: RT:10000:2 OSPF ROUTER ID:172.17.22.22:0
      OSPF RT:0.0.0.0:2:0
    mpls labels in/out nolabel/20012
    rx pathid: 0, tx pathid: 0x0
```

Look at the Next-Hops of VPNv6 learned routes, those are IPv4-IPv6 mapped addresses.

```
CustB-RIP-CE1(config-if)#do sh ipv6 route ospf | begin FD
OI  FD02:172:17:255::2/128 [110/20]
    via FE80::A8BB:CCFF:FE00:300, Ethernet0/0
OI  FD02:172:17:255::3/128 [110/20]
    via FE80::A8BB:CCFF:FE00:300, Ethernet0/0

CustB-RIP-CE2(config)#do sh ipv6 route ospf | begin FD
OI  FD02:172:17:255::1/128 [110/20]
    via FE80::A8BB:CCFF:FE00:600, Ethernet0/0
OI  FD02:172:17:255::3/128 [110/20]
    via FE80::A8BB:CCFF:FE00:600, Ethernet0/0

CustB-RIP-CE3(config-if)#do sh ipv6 route ospf | begin FD
OI  FD02:172:17:255::1/128 [110/20]
    via FE80::A8BB:CCFF:FE00:510, Ethernet1/0
OI  FD02:172:17:255::2/128 [110/20]
    via FE80::A8BB:CCFF:FE00:510, Ethernet1/0
```

```

CustB-RIP-CE1(config-if)#do ping ipv6 FD02:172:17:255::2 source lo 0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FD02:172:17:255::2, timeout is 2 seconds:
Packet sent with a source address of FD02:172:17:255::1
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
    
```

```

CustB-RIP-CE1(config-if)#do ping ipv6 FD02:172:17:255::3 source lo 0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FD02:172:17:255::3, timeout is 2 seconds:
Packet sent with a source address of FD02:172:17:255::1
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
    
```

```

CustB-RIP-CE1#traceroute
Protocol [ip]: ipv6
Target IPv6 address: FD02:172:17:255::2
Source address: FD02:172:17:255::1
Insert source routing header? [no]:
Numeric display? [no]:
Timeout in seconds [3]:
Probe count [3]:
Minimum Time to Live [1]:
Maximum Time to Live [30]:
Priority [0]:
Port Number [0]:
Type escape sequence to abort.
Tracing the route to FD02:172:17:255::2

 1 FD02:172:17:11::11 1 msec 1 msec 0 msec
 2 ::FFFF:10.99.11.99 [MPLS: Labels 90002/20012 Exp 0] 2 msec 2 msec 1 msec
    
```

```

3 FD02:172:17:22::22 [MPLS: Label 20012 Exp 0] 1 msec 1 msec 1 msec
4 FD02:172:17:22::2 2 msec 1 msec 1 msec
    
```

In the SP Core we have not enabled IPv6 unicast routing, Only PE-CE connection is IPv6 unicast using VRF, this method is called 6VPE MPLS VPN. Take a look at below packet capture:

23	13.703792281	10.255.255.99	10.255.255.11	TCP	60	59202 → 646 [ACK] Seq=19 Ack=19 Win=3804 Len=0
24	13.704213346	aa:bb:cc:00:03:30	ISIS-all-level-2-IS...	ISIS C...	292	L2 CSNP, Source-ID: 0000.0000.0011, Start LSP-ID: 0000.0000.0000.00-00, t
25	13.908227112	10.99.11.99	224.0.0.2	LDP	76	Hello Message
26	14.805673724	aa:bb:cc:00:03:30	aa:bb:cc:00:03:30	LOOP	60	Reply
27	14.824802134	10.255.255.99	10.255.255.11	LDP	76	Hello Message
28	15.317557589	10.99.11.11	224.0.0.2	LDP	76	Hello Message
29	15.412116358	fd02:172:17:255::1	fd02:172:17:255::2	ICMPv6	122	Echo (ping) request id=0x0eb4, seq=0, hop limit=63 (no response found!)
30	15.414745077	fd02:172:17:255::1	fd02:172:17:255::2	ICMPv6	122	Echo (ping) request id=0x0eb4, seq=1, hop limit=63 (no response found!)
31	15.416099529	fd02:172:17:255::1	fd02:172:17:255::2	ICMPv6	122	Echo (ping) request id=0x0eb4, seq=2, hop limit=63 (no response found!)
32	15.418058304	fd02:172:17:255::1	fd02:172:17:255::2	ICMPv6	122	Echo (ping) request id=0x0eb4, seq=3, hop limit=63 (no response found!)
33	15.419552268	fd02:172:17:255::1	fd02:172:17:255::2	ICMPv6	122	Echo (ping) request id=0x0eb4, seq=4, hop limit=63 (no response found!)
34	15.697122322	aa:bb:cc:00:03:30	ISIS-all-level-2-IS...	ISIS H...	1514	L2 HELLO, System-ID: 0000.0000.0011

▶	Frame 29: 122 bytes on wire (976 bits), 122 bytes captured (976 bits) on interface 0
▶	Ethernet II, Src: aa:bb:cc:00:03:30 (aa:bb:cc:00:03:30), Dst: aa:bb:cc:00:02:30 (aa:bb:cc:00:02:30)
▶	MultiProtocol Label Switching Header, Label: 90002, Exp: 0, S: 0, TTL: 63
▶	MultiProtocol Label Switching Header, Label: 20012, Exp: 0, S: 1, TTL: 63
▶	Internet Protocol Version 6, Src: fd02:172:17:255::1, Dst: fd02:172:17:255::2
▶	Internet Control Message Protocol v6